УНІВЕРСИТЕТ СПОЖИВЧОЇ КООПЕРАЦІЇ УКРАЇНИ
ПОЛТАВА

О.О. ЄМЕЦЬ
О.В. РОСКЛАДКА

ЗАДАЧІ ОПТИМІЗАЦІЇ НА ПОЛІКОМБІНАТОРНИХ МНОЖИНАХ: властивості та розв'язування

Монографія

УКООПСПІЛКА
ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ
СПОЖИВЧОЇ КООПЕРАЦІЇ УКРАЇНИ
Кафедра математичного моделювання
та соціальної інформатики

О.О. ЄМЕЦЬ
О.В. РОСКЛАДКА

ЗАДАЧІ ОПТИМІЗАЦІЇ НА ПОЛІКОМБІНАТОРНИХ МНОЖИНАХ:
властивості та розв'язування

МОНОГРАФІЯ

ПОЛТАВА
РВЦ ПУСКУ
2006

УДК 519.7:512 11
ББК 22.18
Є60

Рекомендовано до друку вченою радою Полтавського університету споживчої кооперації України від 20 квітня 2006 р.

Рецензенти: ***Смеляков С.В.***, *професор кафедри математичного та програмного забезпечення АСУ Харківського університету Повітряних Сил ім. Івана Кожедуба, д.ф.-м.н.;*
Руткас А.Г., *завідувач кафедри моделювання та математичного забезпечення НОМ Харківського національного університету ім. В.Н. Каразіна, професор, д.ф.-м.н.*

Ємець О.О., Роскладка О.В.
Є60 Задачі оптимізації на полікомбінаторних множинах: властивості та розв'язування: Монографія. – Полтава: РВЦ ПУСКУ, 2006. – 129 с.

ISBN 966-7971-39-2

У монографії розглядаються полікомбінаторні множини, їх опуклі оболонки та задачі оптимізації на них.

Для многогранників поліпереставлень і полірозміщень доведені властивості невиродженості та еквівалентності.

Розв'язана задача розміщення об'єктів обслуговування як задача евклідової полікомбінаторної оптимізації. Для її розв'язування застосовано метод гілок і меж та метод динамічного програмування. Проведено аналіз застосованих алгоритмів та доведена їх ефективність.

Для студентів спеціальності «Соціальна інформатика», «Інформатика», аспірантів і широкого кола фахівців, які цікавляться математичним моделюванням та теорією комбінаторної оптимізації.

УДК 519.7:512 11
ББК 22.18

ISBN 966-7971-39-2

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

Позначення	Що означає
$convM$	опукла оболонка множини M
$dim\,M$	вимірність опуклої оболонки множини M
$\gamma(M)$	кількість гіперграней, що перетинаються в одній вершині многогранника M
G	мультимножина
$S(G)$	основа мультимножини G
$[G]$	первинна специфікація мультимножини G
$\|G\|$	кількість елементів у мультимножині G
$E_{kn}(G)$	загальна множина переставлень елементів мультимножини G, що містить k елементів, серед яких n різних
$E_{\eta n}^{k}(G)$	загальна множина розміщень по k елементів з мультимножини η елементів, серед яких n різних
$E_{kn}^{s}(G)$	множина поліпереставлень елементів мультимножини, що містить k елементів, серед яких n різних і утворена розбиттям мультимножини G на s підмножин
H	множина номерів елементів мультимножини G при утворенні полікомбінаторної множини
$E_{\eta n}^{ks}(G,H)$	множина полірозміщень по k елементів з мультимножини η елементів, серед яких n різних, що утворена розбиттям мультимножини на s підмножин
$E_{\eta n}^{ks}(G,H,\Xi)$	загальна полікомбінаторна множина із впорядкованим набором умов Ξ
J_n	множина перших n натуральних чисел
J_n^0	множина перших n натуральних чисел з нулем
R^n	n-вимірний евклідів простір
$\Pi_{kn}(G)$	загальний переставний многогранник
$\Pi_{\eta n}^{k}(G)$	загальний многогранник розміщень
$\Pi_{kn}^{s}(G,H)$	загальний поліпереставний многогранник
$\Pi_{\eta n}^{ks}(G,H)$	загальний многогранник полірозміщень

ЗМІСТ

ВСТУП

Дослідження задач дискретної оптимізації є передумовою успішного моделювання важливих економічних, природних, соціальних та інших процесів. Методи дискретного програмування є менш розробленими, ніж методи лінійного та опуклого програмування. Однак теорія дискретного програмування неперервно вдосконалюється, а її методи постійно розвиваються. Дослідження в області дискретної оптимізації проводяться в провідних наукових центрах. Велика кількість публікацій, що з'явилася останнім часом і присвячена дискретній та комбінаторній оптимізації, свідчить про необхідність та важливість подібних досліджень. В Україні серед вчених, роботи яких присвячені різним аспектам комбінаторної оптимізації, в першу чергу, слід виділити І.В. Сергієнка, Н.З. Шора (Інститут кібернетики НАН України), Ю.Г. Стояна (Інститут проблем машинобудування НАН України), О.О. Ємця (Полтавський університет споживчої кооперації України), О.А. Павлова (Національний технічний університет України (КПІ)), В.О. Перепелицю (Запорізький національний університет), С.В. Яковлєва (Національний університет внутрішніх справ, м. Харків) та багатьох інших [1–153].

Оптимізаційні задачі комбінаторного типу складають важливий клас дискретних оптимізаційних задач, при розв'язуванні яких знаходиться екстремальне значення цільової функції, заданої на деякій комбінаторній множині (переставлень, розміщень, сполучень тощо.)

Наступним кроком у розв'язанні задач на комбінаторних множинах стало виокремлення евклідової комбінаторної оптимізації – нового напряму, який швидко розвивається [20–33; 55–119].

Теорія і методи евклідової комбінаторної оптимізації включають систематичне вивчення властивостей евклідових комбінаторних множин та їх системне дослідження. Поряд з добре відомими евклідовими комбінаторними множинами переставлень, розміщень, сполучень, розбиттів виділяються більш складні структури – полікомбінаторні множини. Інтерес до них обумовлений, перш за все, потребами практики, оскільки значна кількість задач (напр. [71; 80; 81; 86]) добре описується саме з використанням апарату полікомбінаторних множин [24; 55–58; 61; 63; 68; 115; 116]. Для дослідження властивостей задач на полікомбінаторних множинах необхідно визначити і дослідити властивості цих комбінаторних множин та їх опуклих оболонок.

Тому задачі евклідової оптимізації на полікомбінаторних множинах невідривно пов'язані з комбінаторною теорією многогранників. А отже, дослідження властивостей комбінаторних многогранників, що є опуклими оболонками евклідових комбінаторних множин, є необхідним для пошуку розв'язків оптимізаційних задач.

Дослідження останніх років в області комп'ютерних технологій обумовлюють підвищений інтерес до комбінаторних і полікомбінаторних конфігурацій, які використовуються при створенні сучасних алгоритмів і програм розв'язування оптимізаційних задач на ЕОМ.

Таким чином, все викладене вище, і найголовніше – нові потреби практики, підкреслюють актуальність нових досліджень в області оптимізації на полікомбінаторних множинах.

РОЗДІЛ 1
ЗАГАЛЬНІ ВІДОМОСТІ ПРО ЗАДАЧІ ЕВКЛІДОВОЇ КОМБІНАТОРНОЇ ОПТИМІЗАЦІЇ

1.1. Основи теорії комбінаторної оптимізації

1.1.1. Розвиток теорії комбінаторної оптимізації та комбінаторних методів розв'язування оптимізаційних задач. Велика кількість різноманітних задач планування, проектування, організації виробництва, керування формалізуються за допомогою моделей, в яких значення параметрів обираються з деякої дискретної сукупності заданих величин.

Зародившись у середині минулого століття, дискретне програмування швидко привернуло увагу багатьох вчених в усьому світі, насамперед, завдяки актуальності задач з дискретними параметрами. Відсутність опуклості та зв'язності області допустимих розв'язків задачі дискретної оптимізації зробила неможливим застосування відомих на той час стандартних прийомів математичного програмування. Це призвело до появи нових методів дискретної оптимізації, основні з яких можна об'єднати у дві великі групи: методи відсікання та комбінаторні методи.

Основна ідея методів відсікання була запропонована Данцигом [154] у 1957 р. і була реалізована у роботах Р. Гоморі [155; 156]. Різні способи побудови відсікань для пошуку розв'язку дискретної задачі дали інші алгоритми, ідейно близькі до першого алгоритму Гоморі. Таким чином, були отримані другий алгоритм Гоморі для частково цілочислової задачі, алгоритм Дальтона і Ллевеліна для частково дискретної задачі [157], алгоритм Фінкельштейна [158] для задачі лінійного програмування з булівими змінними та інші. Методи відсікання з часом не втратили своєї актуальності і зараз використовуються для різних класів задач дискретної оптимізації [41; 65; 69; 87; 92; 115; 116; 159].

Важливе місце серед дискретних задач займають задачі комбінаторного типу. Скінченність множини допустимих розв'язків задачі та її комбінаторний характер лягли в основу великої групи комбінаторних методів. Ідею використання направленого перебору допустимих розв'язків запропонували Ленд і Дойг [160] в 1960 році. Пізніше цей підхід був використаний в одному з алгоритмів комбінаторної оптимізації – методі гілок і меж [132; 133]. Крім цього, до поширених комбінаторних методів відносять [132] також адитивний алгоритм Балаша [161], метод послідовного аналізу варіантів [140], метод локальної оптимізації [1; 2; 132] та інші. Окремо слід виділити метод динамічного програмування [136–139], для якого характерним є багатокроковий підхід до розв'язування задачі.

Осторонь вказаних методів стоять евристичні алгоритми напр., [1; 2; 162]. Як правило, їх використовують для швидкого отримання допустимого розв'язку оптимізаційної задачі, нехтуючи можливістю його неоптимальності.

Поряд з очевидними перевагами комбінаторних методів розв'язування оптимізаційних задач кожен з них має свої недоліки, які в певних умовах можуть виявитися суттєвими. Так, складність методу гілок і меж значною мірою залежить від специфіки задачі, що розглядається, від вибраного дослідником способу розбиття множини допустимих розв'язків на підмножини (галуження) і від оцінок та способу їх обчислення.

Метод динамічного програмування має відносно вузькі рамки застосування, оскільки вимагає строгого виконання необхідних умов, які дозволяють розглядати модель комбінаторної задачі в якості динамічної системи. Принцип оптимальності Р. Беллмана [136], який є фундаментальним принципом динамічного програмування є дуже загальним і не дає рекомендації про засоби розв'язування підзадач, що виникають на кожному кроці. Саме тому не існує універсального алгоритму динамічного програмування.

Основні дослідження в рамках дискретної оптимізації ведуться в області цілочислової оптимізації [1–4; 15–18; 32; 34; 35; 37], векторної оптимізації [9; 11; 13; 35; 39; 40], комбінаторної оптимізації та дослідження стійкості різних типів дискретних задач [5; 7–9; 49–51; 54]. Зокрема, комбінаторна оптимізація розвивається в рамках дослідження задач геометричного проектування [20–30; 66; 67; 120], у напрямі оптимізації на комбінаторних множинах [2; 14; 31; 36; 54–119; 122–128; 145; 148; 150; 151], дослідження комбінаторних многогранників [38; 129; 141; 144; 149; 152; 153], дослідження складності комбінаторних задач [44–48; 141; 153], дослідження генетичних алгоритмів [52; 144; 146] тощо.

Велика різноманітність та багатогранність методів відсікання і комбінаторних методів дозволяє робити всебічний аналіз задач дискретної оптимізації. Тому комплексне використання різних методів пошуку розв'язку є ефективним інструментом для ґрунтовного дослідження оптимізаційної задачі. Застосуванню такого підходу до розв'язування задачі розміщення об'єктів обслуговування присвячені четвертий і п'ятий розділи монографії.

1.1.2. Деякі основні положення комбінаторної теорії многогранників. Аналіз задачі комбінаторної оптимізації вимагає глибокого дослідження відповідного многогранника – опуклої оболонки множини допустимих розв'язків комбінаторної задачі. Саме тому, поряд з розвитком аналітичних методів комбінаторної оптимізації розвивається і комбінаторна теорія многогранників, яка вивчає екстремальні властивості многогранника, досліджуючи комплекс усіх його граней.

У комбінаторній теорії многогранник може бути представлений в параметричній або аналітичній формі.

При параметричному представленні многогранник задається у вигляді опуклої оболонки скінченої множини його точок – вершин, або граней нульової вимірності. Проблемам побудови опуклих оболонок присвячені зокрема роботи [74; 163–167]. Також з параметричною формою задання, як наголошено в [38], пов'язана головна задача комбінаторної теорії многогранників – класифікація многогранників із заданою структурою його граней. Дослідження цієї проблеми спирається на поняття комбінаторної еквівалентності многогранників. Виявлення класів еквівалентних многогранників дозволяє поширювати відомі комбінаторні властивості на всіх представників класу. Однак, дослідження еквівалентності опуклих оболонок вимагає розв'язання ще однієї проблеми – доведення невиродженості відповідного комбінаторного многогранника.

Розвиток евклідової комбінаторної оптимізації також нерозривно пов'язаний з дослідженням опуклих оболонок евклідових комбінаторних множин. Так, останнім часом досліджені основні властивості загального переставного та поліпереставного многогранників [24; 57, 66; 67; 73; 76], загального многогранника розміщень [24; 66; 67], загального многогранника поліразміщень [28; 115; 116], многогранника сполучень з повтореннями [24; 66; 67] і деякі властивості загального многогранника сполучень [74; 105; 106].

Однак, питання невиродженості та еквівалентності многогранників саме в евклідовій комбінаторній оптимізації поки що не розглядалися. Виокремлення класів еквівалентності та доведення невиродженості опуклих оболонок для деяких евклідових комбінаторних множин розвинуті в другому розділі монографії.

При аналітичному представленні многогранник задається у вигляді множини розв'язків скінченої системи нерівностей, кожна з яких описує гіпергрань максимальної вимірності многогранника. Така форма задання многогранника має велике значення для пошуку розв'язку лінійних дискретних оптимізаційних задач, оскільки дозволяє використовувати методи і алгоритми лінійного програмування.

Початок використання системи лінійних нерівностей для опису многогранників відноситься до середини XIX ст., але систематизація досліджень і зародження теорії систем лінійних нерівностей як окремого напряму, пов'язана з роботами Г.Ф. Вороного [168; 169]. Фундаментальним дослідженням в області систем лінійних нерівностей присвячені також роботи Г. Мінковського [170], Г. Вейля [171], С.Н. Чернікова, Н.Н. Астафьєва [173], В.С. Чаріна [174].

У теорії систем лінійних нерівностей, що описують многогранники, важливим є поняття незвідної лінійної системи, тобто такої, яка не містить надлишкових обмежень. Питання визначення універсальних критеріїв надлишкових обмежень та формування незвідних лінійних систем розглянуті в роботах [38; 159; 172]. Особливого значення доведення незвідності набуває при розв'язуванні задач на комбінаторних множинах, оскільки з ростом вимірності задачі стрімко зростає кількість обмежень у системі нерівностей. Наявність у такій системі надлишкових обмежень значно підвищує складність обчислень.

Встановленню незвідних систем для опуклих оболонок евклідових комбінаторних множин переставлень у задачах з лінійною та дробово-лінійною цільовими функціями присвячені праці [24; 73; 76; 94; 109; 110; 113; 114].

Для інших евклідових комбінаторних і полікомбінаторних многогранників дослідження незвідності відповідних систем обмежень не проводилися.

Визначенню надлишкових нерівностей в системах, що описують опуклі оболонки загальних евклідових комбінаторної та полікомбінаторної множин розміщень, а також доведенню незвідності систем обмежень відповідних многогранників присвячений третій розділ монографії.

1.2. Вступ в теорію евклідової комбінаторної оптимізації

Розвиток геометричного проектування привів до появи нового напряму досліджень в області дискретної оптимізації і, особливо, дискретних комбінаторних задач. У 80-х роках минулого століття були опубліковані перші роботи з евклідової комбінаторної оптимізації члена-кореспондента НАН України Ю.Г. Стояна [20; 21] – засновника цієї теорії.

Подальшою розробкою теорії і методів евклідової комбінаторної оптимізації разом зі Ю.Г. Стояном займаються О.О. Ємець, С.В. Яковлєв та керовані ними наукові колективи, наприклад, роботи [22–33; 55–119].

Розглянемо елементи теорії евклідової комбінаторної оптимізації, необхідні для викладення результатів досліджень, спираючись, в основному, на [24].

Нехай J_n – множина n перших натуральних чисел, тобто $J_n = \{1,...,n\}$. Мультимножиною:

$$G = \{g_1, g_2, ..., g_\eta\} \tag{1.1}$$

називають сукупність елементів, серед яких можуть бути й однакові. Таким чином, множина – це мультимножина, в якій всі елементи різні. Будь-яку мультимножину G можна задати кортежем всіх її різних елементів – основою, яку позначають $S(G)=(e_1,...,e_n)$, і числом повторень кожного елемента основи – кратністю. Кратності η_i елементів e_i мультимножини записують, як правило, у вигляді степеня, тобто: $G=\left\{e_1^{\eta_1},e_2^{\eta_2},...,e_n^{\eta_n}\right\}$. Кортеж кратностей називають первинною специфікацією та позначають: $[G]$ – кортеж $(\eta_1,\ \eta_2,...,\ \eta_n)=[G]$.

Назвемо *k*-вибіркою підмультимножину мультимножини G, яка містить *k* елементів.

Будемо вважати, що елементи мультимножини G упорядковані:

$$g_j \le g_{j+1} \qquad \forall j \in J_{\eta-1}. \tag{1.2}$$

Розглянемо упорядковану *k*-вибірку з мультимножини G:

$$e=\left(g_{i_1},g_{i_2},...,g_{i_k}\right), \tag{1.3}$$

де $g_{i_j} \in G,\ i_j \ne i_t \forall\ i_j,i_t \in J_\eta,\ \forall\ j,\ t \in J_k$.

Означення 1.1. Множину E, елементами якої є різні упорядковані *k*-вибірки вигляду (1.3) з мультимножини G називають евклідовою комбінаторною множиною (коротко – е-множиною).

Евклідові комбінаторні множини можна відображати в арифметичний евклідовий простір R^k.

Означення 1.2. Нехай E – евклідова комбінаторна множина, а e в зображенні (1.3) – елемент E. Тоді відображення

$$f: E \to E_f \subset R^k$$

називають зануренням E в арифметичний евклідовий простір, якщо між E та E_f існує взаємно однозначна відповідність встановлена правилом:
для

$$e=\left(g_{i_1},g_{i_2},...,g_{i_k}\right)\in E,\ x=f(e),\ x=\left(x_1,x_2,...,x_k\right)\in E_f$$

маємо $x_j = g_{i_j} \quad \forall j \in J_k$.

Множину E_f будемо називають спеціальною комбінаторною множиною [20]. Множина E_f також є евклідовою комбінаторною множиною.

Розглянемо деякі з відомих евклідових комбінаторних множин [24].

Множина переставлень без повторення з k різних дійсних чисел. Нехай G – множина, тобто $[G]=\left(1^n\right)$, $\eta=n=k$, $G=S(G)$. Множину всіх k-вибірок вигляду (1.3) з множини G за таких умов називають множиною переставлень без повторення з k різних дійсних чисел множини G. Позначають цю множину переставлень $P_k(G)$. Образ відображення f, яке кожному переставленню множини $P_k(G)$ ставить у відповідність точку арифметичного евклідового простору R^k, позначають $E_k(G)$, тобто $E_k(G)=f(P_k(G))$.

Множина переставлень з повтореннями з k дійсних чисел, серед яких n різних. Нехай $G=\left\{g_1, g_2,...,g_\eta\right\}$ – мультимножина з основою $S(G)=\left(e_1, e_2,..., e_n\right)$, причому $k=\eta>n$. За таких умов множину всіх k-вибірок вигляду (1.3) з мультимножини G називають множиною переставлень з k дійсних чисел, серед яких n різних, або загальною множиною переставлень. Позначимо цю множину $P_{kn}\left(G\right)$. Образ $f(P_{kn}(G))$ цієї множини в R^k позначимо через $E_{kn}(G)$.

Опуклою оболонкою точок евклідової комбінаторної множини переставлень з повтореннями, як відомо [23], є загальний переставний многогранник $\Pi_{kn}\left(G\right)$, що описується наступною системою лінійних обмежень:

$$\begin{cases} \sum\limits_{j=1}^{i} x_{\alpha_j} \geq \sum\limits_{j=1}^{i} g_j, \\ \sum\limits_{j=1}^{k} x_j = \sum\limits_{j=1}^{k} g_j, \end{cases} \qquad (1.4)$$

де $\alpha_j \in J_k,\ \alpha_j \neq \alpha_t,\ \forall j \neq t,\ j \in J_i,\ t \in J_i,\ \forall i \in J_k$.

Сукупність нерівностей системи (1.4), які мають однакове значення i, називають спілкою i нерівностей цієї системи. Позначимо через η_i $\left(i \in J_n\right)$ кратність елемента основи e_i в мультимножині G.

Теорема 1.3. [58, 60]. Виключення з системи (1.4) при $\eta_1>1$ всіх нерівностей спілок 2, 3, ..., η_1, а при $\eta_n>1$ всіх нерівностей спілок

$k-\eta_n$, $k-\eta_n+1$, ..., $k-2$ перетворює її в незвідну систему обмежень многогранника $\Pi_{kn}(G)$.

Множина k-розміщень без повторення з n різних дійсних чисел. Нехай G – множина, тобто $[G]=(1^n)$, $\eta=n$, $G=S(G)$. Множину всіх k-вибірок вигляду (1.3) з множини G називають множиною k-розміщень без повторення з n різних дійсних чисел множини G. Позначають цю множину розміщень $A_n^k(G)$. Образ f (за означенням 1.2), яке кожному розміщенню множини $A_n^k(G)$ ставить у відповідність точку арифметичного евклідового простору R^k, позначають $E_n^k(G)$, тобто $E_n^k(G)=f(A_n^k(G))$.

Множина k-розміщень з повтореннями з n різних дійсних чисел. Нехай $G=\{g_1,g_2,...,g_\eta\}$ – мультимножина з основою $S(G)=(e_1,e_2,...,e_n)$ та первинною специфікацією $[G]=(k^n)$, тобто кратності η_i елементів e_i однакові і дорівнюють $k\ \forall i\in J_k$, а $\eta=nk$. За таких умов множину всіх упорядкованих k-вибірок вигляду (1.3) з такої мультимножини G називають множиною k-розміщень з необмеженими повтореннями з n різних дійсних чисел. Позначимо цю множину розміщень $\overline{A}_n^k(G)$, а її образ $\overline{E}_n^k(G)=f(\overline{A}_n^k(G))$.

Загальна множина k-розміщень. Нехай $G=\{g_1,g_2,...,g_\eta\}$ – мультимножина з основою $S(G)=(e_1,e_2,...,e_n)$ та первинною специфікацією $[G]=(\eta_1,\eta_2,...,\eta_n)$, де $\eta_i\le k\forall i\in J_k$. Сукупність усіх упорядкованих k-вибірок вигляду (1.3) з мультимножини G будемо називати за таких умов загальною множиною k-розміщень, яку позначають через $A_{\eta n}^k(G)$. Оскільки елементи первинної специфікації мультимножини G задовольняють умови $\eta_i\le k$, то в кожному елементі множини $A_\eta\ (G)$ не більше η_i елементів $e_i\in S(G)\forall i\in J_n$. При $n=\eta$, тобто при $\eta_i=1\forall i\in J_n$, множина $A_{\eta n}(G)$ збігається з множиною $A_n^k(G)$ розміщень без повторень, а при $\eta_i=k\forall i\in J_n$, тобто при $\eta=kn$, множина $A_{\eta n}(G)$ перетворюється на множину $\overline{A}_n^k(G)$.

Загальну множину розміщень $A_{\eta n}(G)$, занурену в евклідів простір R^k позначимо через $E_{\eta n}(G)$.

Опуклу оболонку точок загальної множини розміщень називають загальним многогранником розміщень і позначають $\Pi_{\eta n}^{k}(G) = conv \quad E_{\eta n}^{k}(G)$.

Загальний многогранник розміщень в аналітичному вигляді, як відомо [24], задається наступною системою лінійних нерівностей:

$$\sum_{j=1}^{|\omega|} g_j \le \sum_{i\in\omega} x_i \le \sum_{j=1}^{|\omega|} g_{\eta-j+1}, \ \forall \omega \subset J_k . \qquad (1.5)$$

Многогранник $\Pi_{\eta}^{k}(G)$ містить надлишкові нерівності, тобто є виродженим. Встановлення надлишкових обмежень системи (1.5) та отримання незвідної системи, що описує $\Pi_{\eta}^{k}(G)$, представлено в підрозділі 3.1 монографії.

1.3. Постановки задач евклідової комбінаторної оптимізації

Нехай M − скінчена множина комбінаторної природи, а F − деякий функціонал на цій множині.

Означення 1.4. [24] Основною задачею комбінаторної оптимізації називають задачу знаходження екстремуму

$$F(m^*) = \underset{m\in M}{extr}\, F(m) \qquad (1.6)$$

і екстремалі

$$m^* = arg\, \underset{m\in M}{extr}\, F(m). \qquad (1.7)$$

У залежності від типу екстремуму розглядають основну задачу мінімізації на комбінаторній множині:

$$F(m^*) = \min_{m\in M} F(m),\ m^* = arg \min_{m\in M} F(m),$$

а також основну задачу максимізації:

$$F(m^*) = \max_{m\in M} F(m),\ m^* = arg \max_{m\in M} F(m).$$

Розглянемо клас задач комбінаторної оптимізації, де в ролі множини M виступає евклідова комбінаторна множина E. Для цього класу задача (1.6), (1.7) прийме вигляд:

$$F(g^*) = \underset{g\in E}{extr}\, F(g); \qquad (1.8)$$

$$g^* = arg\, \underset{g\in E}{extr}\, F(g). \qquad (1.9)$$

Введення відображення f дає можливість замінити розв'язок задачі (1.8), (1.9) розв'язком наступної задачі:

$$\Phi(x^*) = \underset{x \in E_\varphi}{extr}\ \Phi(x); \qquad (1.10)$$

$$x^* = arg \underset{x \in E_\varphi}{extr}\ \Phi(x), \qquad (1.11)$$

де $\Phi(x)$ – означена на множині E_φ функція k змінних, яка відповідає функціоналу $F(g), g \in E$.

Означення 1.5. [23] Задачу (1.10), (1.11) називають евклідовою задачею комбінаторної оптимізації.

Часто задачу (1.10), (1.11) зображують у такому вигляді:
знайти

$$\Phi(x^*) = \underset{x \in E_\psi}{extr}\ \Phi(x); \qquad (1.12)$$

$$x^* = arg \underset{x \in E_\psi}{extr}\ \hat{O}(x), \qquad (1.13)$$

при обмеженнях

$$\psi^i(x) \le 0 \qquad \forall i \in J_r; \qquad (1.14)$$

$$\psi^{r+i}(x) = 0 \qquad \forall i \in J_s, \qquad (1.15)$$

де r, s – деякі цілі невід'ємні константи, а множина $E_\psi \subset R^k$ і функції $\psi^i : E_\psi \to R^1 \forall i \in J_{r+s}$ такі, що:

$$E_\varphi = \{x \mid \ x \in E_\psi, \ \ \psi^i(x) \le 0 \qquad \forall i \in J_r; \ \psi^{r+i}(x) = 0 \qquad \forall i \in J_s\}.$$

Обмеження (1.14), (1.15) називають додатковими обмеженнями евклідової комбінаторної задачі. Якщо додаткові обмеження відсутні, тобто $r + s = 0$, то задачу (1.12) – (1.15) називають безумовною задачею евклідової комбінаторної оптимізації, в іншому випадку – умовною евклідовою комбінаторною задачею.

1.4. Основні властивості евклідових полікомбінаторних множин

Моделювання складних задач комбінаторного типу вимагає використання більш складних комбінаторних структур – полікомбінаторних множин. Важливі задачі на полікомбінаторних множинах розглянуті в [24; 28; 61; 63; 66–68; 81; 97; 100; 115; 116].

Розглянемо відомі і досліджені евклідові полікомбінаторні множини [24; 28; 58; 68].

1.4.1. Евклідова комбінаторна множина поліпереставлень. Нехай $k=\eta$. Представимо множину J_k у вигляді упорядкованого розбиття на s непорожніх підмножин $N_1,...,N_s$, які не перерізаються, тобто для яких виконуються умови: $N_i \bigcap N_j = \varnothing$, $N_i \neq \varnothing$, $N_j \neq \varnothing$, $\forall i,j \in J_s$. Нехай H – множина всіх елементів вигляду:

$$\pi = (\pi(1),...,\pi(k)) = (\pi^1,...,\pi^s),$$

де π^i – довільне переставлення елементів множини N_i $\forall i \in J_s$.

Позначимо через G^{N_i} підмультимножину пронумерованої мультимножини G, що складається з тих елементів G, номери яких належать множині N_i:

$$G^{N_i} = \{g_1^{N_i},\ g_2^{N_i},\ ...,\ g_{n_i}^{N_i}\},$$

де $k_i = |N_i|$; $[G^{N_i}] = \left(\eta_1^i,\ \eta_2^i,\ ...,\ \eta_{n_i}^i\right)$; $g_j^{N_i} \le g_{j+1}^{N_i}$ $\forall j \in J_{n_i-1}$, $\forall i \in J_s$.

Множину $P_{kn}^s(G,H) = \{(g_{\pi(1)},...,g_{\pi(k)}) \mid g_{\pi(i)} \in G \quad \forall i \in J_k,\ \forall \pi \in H\}$ називають загальною поліпереставною множиною. Образ поліпереставної множини в арифметичному евклідовому просторі позначають через $E_{kn}^s(G,H)$, тобто $E_{kn}^s(G,H) = f(P_{kn}^s(G,H))$.

Опуклою оболонкою множини поліпереставлень є загальний поліпереставний многогранник, який, як відомо [24], має вигляд

$$\begin{cases} \sum\limits_{j \in N_i'} x_j = \sum\limits_{j=1}^{k_i} g_j^{N_i} & \forall i \in J_s, \\ \sum\limits_{j \in \omega^i} x_j \ge \sum\limits_{j=1}^{|\omega^i|} g_j^{N_i} & \forall \omega^i \in N_i', \quad \forall i \in J_s, \end{cases} \tag{1.16}$$

де $g_j^{N_i} \in G^{N_i}$, $N_i' = \left\{ \left(\sum\limits_{j=1}^{i-1} k_j\right) + 1, \left(\sum\limits_{j=1}^{i-1} k_j\right) + 2, \ldots, \sum\limits_{j=1}^{i} k_j \right\}$ $\forall i \in J_s$.

Сукупність нерівностей системи (1.16) із фіксованими значеннями пари чисел i та $|\omega^i|$ $\forall \omega^i \subset N_i'$ $\forall i \in J_s$, називають спілкою $\left(i,\ |\omega^i|\right)$ нерівностей цієї системи.

Теорема 1.6. [76]. Виключення з системи (1.16) при $\eta_1^i > 1$ всіх нерівностей спілок $(i,2),(i,3),...,(i,\eta_1^i)$, при $\eta_{n_i}^i > 1$ – всіх нерівностей спілок $\left(i,k_i-\eta'_{n_i}\right),\left(i,k_i-\eta'_{n_i}+1\right),\ldots,\left(i,k_i-2\right)$ та всіх нерівностей спілок $\left(i,k_i\right)$ $\forall i\in J_s$ перетворює систему обмежень многогранника $\Pi^s_{kn}\left(G,H\right)$ в незвідну.

1.4.2. Евклідова комбінаторна множина полірозміщень. Нехай $G=\left\{g_1,g_2,\ldots g_\eta\right\}$ – мультимножина з основою $S(G)=\left(e_1,e_2,\ldots,e_n\right)$ та первинною специфікацією $\left[G\right]=\left(\eta_1,\eta_2,\ldots,\eta_n\right)$, де $1\le\eta_i\le k$ $\forall i\in J_n$. Розглянемо упорядковане розбиття J_η на s множин $N_1,\ldots,N_s$, для яких $N_i\bigcap N_j=\varnothing,\ N_i\ne\varnothing,\ N_j\ne\varnothing$ $\forall i,j\in J$, і упорядковане розбиття константи k на s доданків $k_1,\ldots,k_s$, які задовольняють умову $1\le k_i\le n_i$, де $n_i=\left|N_i\right|$ $\forall i\in J_s$. Очевидно, що $\sum_{i=1}^{s}n_i=\eta$, $\sum_{i=1}^{s}k_i=k$, $s\le k$.

Нехай H – множина k-вибірок з множини J_η вигляду

$$\pi=(\pi(1),\ldots,\pi(k))=(\pi_{11},\ldots,\pi_{1k_1},\ldots,\pi_{s1},\ldots,\pi_{sk_s})=(\pi^1,\ldots,\pi^s),$$

де $\pi^i=(\pi_{i1},\ldots,\pi_{ik_i})$ – довільна k_i-вибірка з множини N_i $\forall i\in J_s$.

Множину $A^{ks}_{\eta n}(G,H)=\{(g_{\pi(1)},\ldots,g_{\pi(k)})\mid\forall\pi\in H\}$ називають загальною множиною полірозміщень, а образ $f(A^{ks}_{\eta n}(G,H))$ позначають через $E^{ks}_{\eta n}(G,H)$.

Загальний многогранник полірозміщень, як відомо [27], задається системою:

$$\sum_{j=1}^{|\omega^i|}g^{N_i}_j\le\sum_{j\in\omega^i}x_j\le\sum_{j=1}^{|\omega^i|}g^{N_i}_{n_i-j+1}\quad\forall\omega^i\subset N'_i\quad\forall i\in J_s,\ (1.17),$$

де $N'_i=\left\{\left(\sum_{j=1}^{i-1}k_j\right)+1,\left(\sum_{j=1}^{i-1}k_j\right)+2,\ldots,\sum_{j=1}^{i}k_j\right\}$ $\forall i\in J_s$;

$g^{N_i}_j\le g^{N_i}_{j+1}$ $\forall j\in J_{n_i-1}$, $\forall i\in J_s$.

Дослідження незвідності системи обмежень загального многогранника полірозміщень до цих пір не були проведені. Встановлення спілок, що містять надлишкові обмеження і отримання незвідної системи для системи (1.17) розглянуто в третьому розділі монографії (підрозділ 3.2).

1.4.3. Приклад утворення евклідової полікомбінаторної множини. Нехай дана мультимножина $G = \{1, 1, 2, 5, 5, 5, 9\}$. Тоді, основа $S(G) = (1, 2, 5, 9)$; первинна специфікація $[G] = (2, 1, 3, 1)$; $\eta = 7$; $n = 4$.

Нехай вимірність простору $k = 5$. Розіб'ємо множину $J_\eta = J_7$ на три підмножини: $N_1 = \{1, 3, 4\}$; $N_2 = \{7\}$; $N_3 = \{2, 5, 6\}$. Таким чином, $s = 3; n_1 = 3; n_2 = 1; n_3 = 3$.

Якщо формувати множину поліпереставлень, то множина H матиме наступний вигляд:
H={(1, 3, 4, 7, 2, 5, 6), (1, 4, 3, 7, 2, 5, 6), (3, 1, 4, 7, 2, 5, 6), (3, 4, 1, 7, 2, 5, 6), (4, 3, 1, 7, 2, 5, 6), (4, 1, 3, 7, 2, 5, 6);
(1, 3, 4, 7, 2, 5, 6), (1, 3, 4, 7, 2, 6, 5), (1, 3, 4, 7, 5, 6, 2), (1, 3, 4, 7, 5, 2, 6), (1, 3, 4, 7, 6, 5, 2), (1, 3, 4, 7, 6, 2, 5);
(1, 4, 3, 7, 2, 5, 6), (1, 4, 3, 7, 2, 6, 5), (1, 4, 3, 7, 5, 6, 2), (1, 4, 3, 7, 5, 2, 6), (1, 4, 3, 7, 6, 5, 2), (1, 4, 3, 7, 6, 2, 5);
(3, 1, 4, 7, 2, 5, 6), (3, 1, 4, 7, 2, 6, 5), (3, 1, 4, 7, 5, 6, 2), (3, 1, 4, 7, 5, 2, 6), (3, 1, 4, 7, 6, 5, 2), (3, 1, 4, 7, 6, 2, 5);
(3, 4, 1, 7, 2, 5, 6), (3, 4, 1, 7, 2, 6, 5), (3, 4, 1, 7, 5, 6, 2), (3, 4, 1, 7, 5, 2, 6), (3, 4, 1, 7, 6, 5, 2), (3, 4, 1, 7, 6, 2, 5);
(4, 3, 1, 7, 2, 5, 6), (4, 3, 1, 7, 2, 6, 5), (4, 3, 1, 7, 5, 6, 2), (4, 3, 1, 7, 5, 2, 6), (4, 3, 1, 7, 6, 5, 2), (4, 3, 1, 7, 6, 2, 5);
(4, 1, 3, 7, 2, 5, 6), (4, 1, 3, 7, 2, 6, 5), (4, 1, 3, 7, 5, 6, 2), (4, 1, 3, 7, 5, 2, 6), (4, 1, 3, 7, 6, 5, 2), (4, 1, 3, 7, 6, 2, 5)}.

Відповідна множина поліпереставлень являтиме собою множину 7-вибірок з мультимножини G з номерами елементів, що відповідають утвореній множині H.
$P^s_{\eta n}(G,\ H) = P^3_{7,\,4}(G,\ H)$={(1, 2, 5, 9, 1, 5, 5), (1, 5, 2, 9, 1, 5, 5), (2, 1, 5, 9, 1, 5, 5), (2, 5, 1, 9, 1, 5, 5), (5, 2, 1, 9, 1, 5, 5), (5, 1, 2, 9, 1, 5, 5);
(1, 2, 5, 9, 1, 5, 5), (1, 2, 5, 9, 1, 5, 5), (1, 2, 5, 9, 5, 5, 1), (1, 2, 5, 9, 5, 1, 5), (1, 2, 5, 9, 5, 5, 1), (1, 2, 5, 9, 5, 1, 5);
(1, 5, 2, 9, 1, 5, 5), (1, 5, 2, 9, 1, 5, 5), (1, 5, 2, 9, 5, 5, 1), (1, 5, 2, 9, 5, 1, 5), (1, 5, 2, 9, 5, 5, 1), (1, 5, 2, 9, 5, 1, 5);
(2, 1, 5, 9, 1, 5, 5), (2, 1, 5, 9, 1, 5, 5), (2, 1, 5, 9, 5, 5, 1), (2, 1, 5, 9, 5, 1, 5), (2, 1, 5, 9, 5, 5, 1), (2, 1, 5, 9, 5, 1, 5);
(2, 5, 1, 9, 1, 5, 5), (2, 5, 1, 9, 1, 5, 5), (2, 5, 1, 9, 5, 5, 1), (2, 5, 1, 9, 5, 1, 5), (2, 5, 1, 9, 5, 5, 1), (2, 5, 1, 9, 5, 1, 5);

(5, 2, 1, 9, 1, 5, 5), (5, 2, 1, 9, 1, 5, 5), (5, 2, 1, 9, 5, 5, 1), (5, 2, 1, 9, 5, 1, 5), (5, 2, 1, 9, 5, 5, 1), (5, 2, 1, 9, 5, 1, 5);
(5, 1, 2, 9, 1, 5, 5), (5, 1, 2, 9, 1, 5, 5), (5, 1, 2, 9, 5, 5, 1), (5, 1, 2, 9, 5, 1, 5), (5, 1, 2, 9, 5, 5, 1), (5, 1, 2, 9, 5, 1, 5)}.

Після вилучення однакових елементів будемо мати
$P^3_{7,4}(G,\ H)$={(1, 2, 5, 9, 1, 5, 5), (1, 5, 2, 9, 1, 5, 5), (2, 1, 5, 9, 1, 5, 5), (2, 5, 1, 9, 1, 5, 5), (5, 2, 1, 9, 1, 5, 5), (5, 1, 2, 9, 1, 5, 5), (1, 2, 5, 9, 5, 5, 1), (1, 5, 2, 9, 5, 5, 1), (2, 1, 5, 9, 5, 5, 1), (2, 5, 1, 9, 5, 5, 1), (5, 2, 1, 9, 5, 5, 1), (5, 1, 2, 9, 5, 5, 1), (1, 2, 5, 9, 5, 1, 5), (1, 5, 2, 9, 5, 1, 5), (2, 1, 5, 9, 5, 1, 5), (2, 5, 1, 9, 5, 1, 5), (5, 2, 1, 9, 5, 1, 5), (5, 1, 2, 9, 5, 1, 5)}.

Якщо формувати множину поліроміщень, то необхідно розбити число k на s доданків. Нехай $k_1 = 2$; $k\ = 1$; $k_3 = 2$. Тоді, множина H матиме наступний вигляд:
H = {(1, 3, 7, 2, 5), (1, 3, 7, 5, 2), (3, 1, 7, 2, 5), (3, 1, 7, 5, 2), (1, 3, 7, 2, 6), (1, 3, 7, 6, 2), (3, 1, 7, 2, 6), (3, 1, 7, 6, 2), (1, 3, 7, 5, 6), (1, 3, 7, 6, 5), (3, 1, 7, 5, 6), (3, 1, 7, 6, 5);
(1, 4, 7, 2, 5), (4, 1, 7, 2, 5), (1, 4, 7, 5, 2), (4, 1, 7, 5, 2), (1, 4, 7, 2, 6), (4, 1, 7, 2, 6), (1, 4, 7, 6, 2), (4, 1, 7, 6, 2), (1, 4, 7, 5, 6), (4, 1, 7, 5, 6), (1, 4, 7, 6, 5), (4, 1, 7, 6, 5);
(3, 4, 7, 2, 5), (4, 3, 7, 2, 5), (3, 4, 7, 5, 2), (4, 3, 7, 5, 2), (3, 4, 7, 2, 6), (4, 3, 7, 2, 6), (3, 4, 7, 6, 2), (4, 3, 7, 6, 2), (3, 4, 7, 5, 6), (4, 3, 7, 5, 6), (3, 4, 7, 6, 5), (4, 3, 7, 6, 5)}.

Відповідна множина поліроміщень матиме такий вигляд:
$E^{ks}_{\eta n}(G,\ H) = E^{53}_{74}(G,\ H)$ = {(1, 2, 9, 1, 5), (1, 2, 9, 5, 1), (2, 1, 9, 1, 5), (2, 1, 9, 5, 1), (1, 2, 9, 1, 5), (1, 2, 9, 5, 1), (2, 1, 9, 1, 5), (2, 1, 9, 5, 1), (1, 2, 9, 5, 5), (1, 2, 9, 5, 5), (2, 1, 9, 5, 5), (2, 1, 9, 5, 5);
(1, 5, 9, 1, 5), (5, 1, 9, 1, 5), (1, 5, 9, 5, 1), (5, 1, 9, 5, 1), (1, 5, 9, 1, 5), (5, 1, 9, 1, 5), (1, 5, 9, 5, 1), (5, 1, 9, 5, 1), (1, 5, 9, 5, 5), (5, 1, 9, 5, 5), (1, 5, 9, 5, 5), (5, 1, 9, 5, 5);
(2, 5, 9, 1, 5), (5, 2, 9, 1, 5), (2, 5, 9, 5, 1), (5, 2, 9, 5, 1), (2, 5, 9, 1, 5), (5, 2, 9, 1, 5), (2, 5, 9, 5, 1), (5, 2, 9, 5, 1), (2, 5, 9, 5, 5), (5, 2, 9, 5, 5), (2, 5, 9, 5, 5), (5, 2, 9, 5, 5) }.

Після вилучення однакових елементів отримаємо:
$E^{53}_{74}(G,\ H)$={(1, 2, 9, 1, 5), (1, 2, 9, 5, 1), (2, 1, 9, 1, 5), (2, 1, 9, 5, 1), (1, 2, 9, 5, 5), (2, 1, 9, 5, 5), (1, 5, 9, 1, 5), (5, 1, 9, 1, 5), (1, 5, 9, 5, 1), (5, 1, 9, 5, 1), (1, 5, 9, 5, 5), (5, 1, 9, 5, 5), (2, 5, 9, 1, 5), (5, 2, 9, 1, 5), (2, 5, 9, 5, 1), (5, 2, 9, 5, 1), (2, 5, 9, 5, 5), (5, 2, 9, 5, 5)}.

1.5. Постановка основної задачі евклідової полікомбінаторної оптимізації

Розглянемо загальну евклідову полікомбінаторну множину в арифметичному евклідовому просторі R^k. Нехай, як і раніше, $G=\{g_1, g_2,...,g_\eta\}$ – мультимножина з основою $S(G)=(e_1, e_2,..., e_n)$ та первинною специфікацією $[G]=(\eta_1, \eta_2,...,\eta_n)$, де $1\le\eta_i\le k \quad \forall i\in J_n$. Оберемо довільним чином натуральне число $s\le k$. Розглянемо упорядковане розбиття множини J_η на s множин $N_1,...,N_s$, для яких $N_i\cap N_j=\varnothing, \quad N_i\ne\varnothing, \; N_j\ne\varnothing \qquad \forall i,j\in J_s$, і упорядковане розбиття константи k на s доданків $k_1,...,k_s$, які задовольняють умову $1\le k_i\le n_i$, де $n_i=|N_i| \quad \forall i\in J_s$.

Нехай H – множина k-вибірок з множини J_η вигляду

$$\pi=(\pi(1),...,\pi(k))=(\pi_{11},...,\pi_{1k_1},...,\pi_{s1},...,\pi_{sk_s})=(\pi^1,...,\pi^s),$$

де $\pi^i=(\pi_{i1},...,\pi_{ik_i})$ – k_i-вибірка з множини $N_i \quad \forall i\in J_s$, але не довільна, а така, що задовольняє певним умовам: $\pi^i\in\Xi_i \quad \forall i\in J_s$.

Позначимо через $\Xi=(\Xi_1, \; ..., \; \Xi_i, \; ..., \; \Xi_s)$ – впорядкований набір умов $\Xi_i \quad \forall i\in J$. Тоді, множину

$$A_{\eta n}^{ks}(G,H,\Xi)=\{(g_{\pi(1)},...,g_{\pi(k)}) \mid \quad \forall\pi\in H, \quad \pi^i\in\Xi_i \quad \forall i\in J_s\}$$

називають [68] загальною полікомбінаторною множиною.

Елементи загальної полікомбінаторної множини називають k-полівибірками або просто полівибірками. Іноді для спрощення будемо позначати загальну полікомбінаторну множину $A_{\eta n}^{ks}(G,H,\Xi)$ через $A(G, H, \Xi)$ або просто через A. Очевидно, що якщо $\pi^i\in\Xi_i$ – довільне переставлення елементів множини $N_i \quad \forall i\in J_s$, то маємо частинний випадок множини $A_\eta^s(G,H,\Xi)$ – розглянуту вище множину поліпереставлень $P_{kn}^s(G, \; H)$. У випадку, якщо $\pi^i\in\Xi_i$ – довільна k_i-вибірка з множини $N_i \quad \forall i\in J_s$, то загальна полікомбінаторна множина $A_\eta^s(G,H,\Xi)$ є множиною полірозміщень $A_{\eta n}^{ks}(G,H)$.

При такому формуванні загальна полікомбінаторна множина $A_{\eta n}^{ks}(G,\ H,\ \Xi)$ задовольняє означення 1.1, тобто є евклідовою комбінаторною множиною. Після занурення її в арифметичний евклідів простір її образ будемо позначати через $E_{\eta n}^{ks}(G,\ H,\ \Xi)$, тобто $E_{\eta n}^{ks}(G,\ H,\ \Xi)=f\left(A_{\eta n}^{ks}(G,\ H,\ \Xi)\right)$ або $E(G,\ H,\ \Xi)=f(A(G,\ H,\ \Xi))$.

Розглянемо декартовий добуток двох загальних евклідових полікомбінаторних множин $E(G,\ H,\ \Xi)=E_1(G_1,\ H_1,\ \Xi_1)\times E_2(G_2,\ H_2,\ \Xi_2)$. Очевидно, що при цьому $E(G,\ H,\ \Xi)$ також є загальною евклідовою полікомбінаторною множиною.

Під основною задачею полікомбінаторної оптимізації будемо розуміти задачу знаходження екстремуму

$$F(x^*)=\underset{x\in E(G,\ H,\ \Xi)}{extr}F(x) \tag{1.18}$$

і екстремалі

$$x^*=arg\underset{x\in E(G,\ H,\ \Xi)}{extr}F(x). \tag{1.19}$$

Далі, всі дослідження задач на евклідових полікомбінаторних множинах в монографії розглядаються як частинні випадки задачі (1.18), (1.19).

РОЗДІЛ 2
ОПУКЛІ ОБОЛОНКИ ЕВКЛІДОВИХ КОМБІНАТОРНИХ МНОЖИН ПЕРЕСТАВЛЕНЬ

2.1. Умови невиродженості загального переставного многогранника

Вивчення властивостей комбінаторних многогранників дозволяє розробляти нові методи і алгоритми розв'язування задач евклідової комбінаторної оптимізації. Важливою задачею при вивченні опуклих оболонок комбінаторних і полікомбінаторних множин є класифікація комбінаторних многогранників.

Розбиття на класи невідривно пов'язане з поняттям еквівалентності многогранників. У свою чергу, властивість еквівалентності визначена лише для невироджених многогранників. Таким чином, визначення умов невиродженості та еквівалентності опуклих оболонок евклідових комбінаторних і полікомбінаторних множин є необхідною передумовою класифікації відповідних многогранників.

Нехай M – многогранник, у кожній вершині якого перетинається однакова кількість гіперграней, яку позначимо через $\gamma(M)$.

Відомі [76] різні вирази для визначення кількості $\gamma(\Pi_{kn}(G))$ гіперграней, які інцидентні довільній вершині загального многогранника переставлень $\Pi_{kn}(G)$, в залежності від основи $S(G)$ мультимножини G. Позначимо через r_i кількість повторень елемента e_i основи в мультимножині G. Так, для $n=2$ знайдено [76, с. 22, 23]:

- якщо $r_1=1,\ r_n>1$, то $\gamma(\Pi_{k,2}(G))=r_n$;
- якщо $r_n=1,\ r_1>1$, то $\gamma(\Pi_{k,2}(G))=r_1$;
- якщо $r_1>1,\ r_n>1$, то $\gamma(\Pi_{k,2}(G))=r_1+r_n$.

Для $n>2$ знайдено [76, с. 23]:

- якщо $r_n=1$, то $\gamma(\Pi_{kn}(G))=r_1+\sum\limits_{i=2}^{n-1}2^{r_i}-(n-2)$;
- якщо $r_n>1,\ r_{n-1}>1$, то $\gamma(\Pi_{kn}(G))=r_1+r_n+\sum\limits_{i=2}^{n-1}2^{r_i}-(n-1)$;
- якщо $r_n>1,\ r_{n-1}=1$, то $\gamma(\Pi_{kn}(G))=r_1+r_n+\sum\limits_{i=2}^{n-2}2^{r_i}-(n-3)$.

Доведемо, що ці вирази при різних значеннях r_n, r_{n-1} та $n>2$ еквівалентні єдиному виразу

$$\gamma\left(\Pi_{kn}(G)\right)=r_1+r_n+\sum_{i=2}^{n-1}2^{r_i}-(n-1). \quad (2.1)$$

Дійсно, при $r_n=1$ маємо $r_1+1+\sum_{i=2}^{n-1}2^{r_i}-(n-1)=r_1+\sum_{i=2}^{n-1}2^{r_i}-(n-2)$.

При $r_{n-1}=1$ маємо:

$$r_1+r_n+\sum_{i=2}^{n-2}2^{r_i}+2^{r_{n-1}}-(n-1)=$$

$$=r_1+r_n+\sum_{i=2}^{n-2}2^{r_i}+2-(n-1)=r_1+r_n+\sum_{i=2}^{n-2}2^{r_i}-(n-3).$$

Означення 2.1. [38, с. 24] Многогранник вимірності d називається простим многогранником, якщо кожна його вершина належить (інцидентна) рівно d граням максимальної вимірності.

Твердження 2.2. [38, с. 37] Многогранник M є виродженим тоді і тільки тоді, коли або M не є простим многогранником, або система, що задає многогранник, містить надлишкові обмеження.

Теорема 2.3. Многогранник $\Pi_{kn}(G)$, що описується незвідною системою (1.4), є невиродженим тоді і тільки тоді, коли:

- $1\in[G]$ при $n=2$;
- $r_i=1 \quad \forall i\in J_{n-1}\setminus\{1\}$ при $n>2$.

Доведення. Достатність. При $n=2$ маємо, що $k=r_1+r_n$. Нехай $1\in[G]$. Тоді можливі три випадки:

- якщо $r_1=1$, то $\gamma\left(\Pi_{k,2}(G)\right)=r_n=k-r_1=k-1$;
- якщо $r_n=1$, то $\gamma\left(\Pi_{k,2}(G)\right)=r_1=k-r_n=k-1$;
- якщо $r_1=1$ і $r_n=1$, то $k=2$, а $\gamma\left(\Pi_{k,2}(G)\right)=1=k-1$.

Нехай $n>2$ і $r_i=1 \quad \forall i\in J_{n-1}\setminus\{1\}$. Тоді за формулою (2.1) маємо:

$$\gamma(\Pi_{kn}(G))=r_1+r_n+\sum_{i=2}^{n-1}2^1-(n-1)=r_1+r_n+2\cdot\sum_{i=2}^{n-1}r_i-(n-1)=$$

$$=\left(r_1+r_n+\sum_{i=2}^{n-1}r_i\right)+\sum_{i=2}^{n-1}r_i-(n-1)=\sum_{i=1}^{n}r_i+(n-2)-(n-1)=\sum_{i=1}^{n}r_i-1=k-1$$

Оскільки $dim\ \Pi_{kn}(G)=k-1$, то за означенням 2.1 многогранник $\Pi_{kn}(G)$ для будь-якого значення n є простим. А оскільки система (1.4) не містить надлишкових обмежень, то за твердженням 2.2 достатність доведено.

Необхідність. Нехай многогранник $\Pi_{kn}(G)$ є невиродженим. З твердження 2.2 випливає, що $\Pi_{kn}(G)$ – простий. Отже, $\gamma(\Pi_{kn}(G))=k-1$.

Доведемо необхідність для $n=2$ від супротивного. Припустимо, що $1\notin[G]$, тобто $r_1>1$ і $r_n>1$. Тоді, $\gamma(\Pi_{kn}(G))=r_1+r_n=k\neq dim\ \Pi_{kn}(G)=k-1$, а це суперечить факту про те, що многогранник $\Pi_{kn}(G)$ є простим. Отже, первинна специфікація мультимножини G для не виродженого многогранника повинна містити 1.

Доведемо необхідність для випадку $n>2$. Використавши (2.1) маємо: $\gamma(\tilde{A}_v)=r_1+r_n+\sum_{i=2}^{n-1}2^{r_i}-(n-1)=k-1$. Оскільки $k=\sum_{i=1}^{n}r_i=r_1+\sum_{i=2}^{n-1}r_i+r_n$, то $r_1+r_n+\sum_{i=2}^{n-1}2^{r_i}-(n-1)=r_1+\sum_{i=2}^{n-1}r_i+r_n-1$,

$$\sum_{i=2}^{n-1}2^{r_i}-(n-1)=\sum_{i=2}^{n-1}r_i-1,$$

$$\sum_{i=2}^{n-1}(2^{r_i}-r_i)=n-2. \qquad (2.2)$$

Проаналізуємо розв'язки рівняння (2.2) відносно змінної r_i. Кількість доданків у лівій частині (2.2) дорівнює $n-1-2+1=n-2$. Очевидно, що $2^{r_i}>r_i$ і різниця $2^{r_i}-r_i>0$ є цілим числом. Тоді маємо, що сума додатних цілих чисел дорівнює їх кількості. Це можливо в єдиному випадку, якщо $r_i=1$. Теорему доведено.

Наприклад, многогранник $\Pi_{7,4}(G)$ є невиродженим, якщо $G=\{3,\ 3,\ 3,\ 5,\ 7,\ 8,\ 8\}$ і виродженим, якщо $G=\{3,\ 3,\ 5,\ 5,\ 7,\ 8,\ 8\}$.

Розглянемо геометричну інтерпретацію для різних випадків мультимножини G.

Приклад 2.4. Нехай $G_1=\{3,5,5,5\}$. Маємо $k=4$, $n=2$, $r_1=1$, $r_n=3$, $dim\ \Pi_{4,2}(G_1)=3$, $\gamma(\Pi_{4,2}(G_1))=r_n=3$. Отже, за теоремою 2.3 переставний многогранник $\Pi_{4,2}(G_1)$ – невироджений (рис. 2.1).

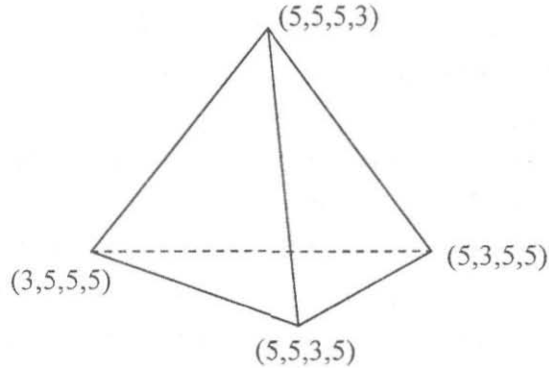

Рис. 2.1

Приклад 2.5. Нехай $G_2=\{3,3,5,5\}$. Маємо $k=4$, $n=2$, $r_1=2$, $r_n=2$, $dim\ \Pi_{4,2}(G_2)=3$, $\gamma(\Pi_{4,2}(G_2))=r_1+r_n=2+2=4$. Отже, за теоремою 2.3. $\Pi_{4,2}(G_2)$ – вироджений (рис. 2.2).

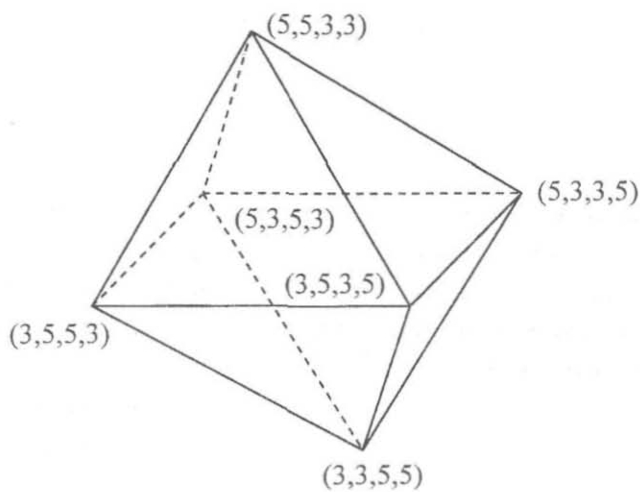

Рис. 2.2

Приклад 2.6. Нехай $G_3=\{3,3,5,7\}$. Маємо $k=4$, $n=3$, $r_1=2$, $r_2=1$, $r_n=1$, $dim\ \Pi_{4,3}(G_3)=3$. За формулою (2.1) $\gamma(\Pi_{4,3}(G_3))=3$. Отже, за теоремою 2.3. $\Pi_{4,3}(G_3)$ – невироджений (рис. 2.3).

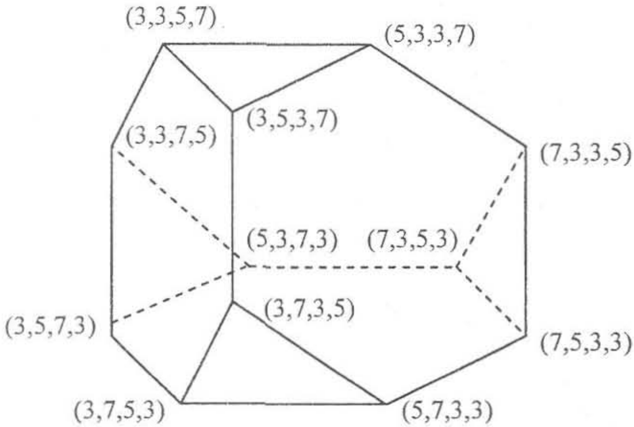

Рис. 2.3

Приклад 2.7. Нехай $G_4=\{3,5,5,7\}$. Маємо $k=4$, $n=3$, $r_1=1$, $r_2=2$, $r_n=1$, $dim\ \Pi_{4,3}(G_4)=3$. За формулою (2.1) $\gamma(\Pi_{4,3}(G_4))=4$. Отже, за теоремою 2.3. $\Pi_{4,3}(G_4)$ – вироджений (рис. 2.4).

Теорема 2.8 (Наслідок з теореми 2.3). Нехай $[G_j^*]\subset[G_j]$ і $r_{ij}^*=r_{ij}\ \forall i\in J_n$, $i\neq 1$, $i\neq n$, де r_{ij}^*, r_{ij} – кількості повторень елемента e_i в мультимножинах G^* і G_j відповідно, $j\in J_2$; $n>2$. Тоді, якщо $\Pi_{kn}(G_1)$ – невироджений многогранник в R^k, що описується незвідною системою вигляду (1.4), і $[G_1^*]=[G_2^*]$, то $\Pi_{kn}(G_2)$ також є невиродженим.

Доведення. Оскільки $\Pi_{kn}(G\ )$ – невироджений, то за теоремою 2.3 для $[G_1]$ маємо: $r_{i1}=1\quad \forall i\in J_{n-1}\setminus\{1\}$, тобто $[G_1^*]=(1,\ 1,\ ...,\ 1)$. Це означає, що первинна специфікація мультимножини G_1 може мати такий вид: $[G_1]=(1,\ 1,\ ...,\ 1)$ або $[G_1]=(\alpha,\ 1,\ ...,\ 1)$, або

$[G_1]=(1,\ 1,\ ...,\ 1,\ \alpha)$, або $[G_1]=(\alpha,\ 1,\ ...,\ 1,\ \beta)$, де $\alpha \neq 1$, $\beta \neq 1$. За умовою $[G_1^*]=[G_2^*]$. Отже, за теоремою 2.3 $\Pi_{kn}(G_2)$ також є невиродженим, що і треба було довести.

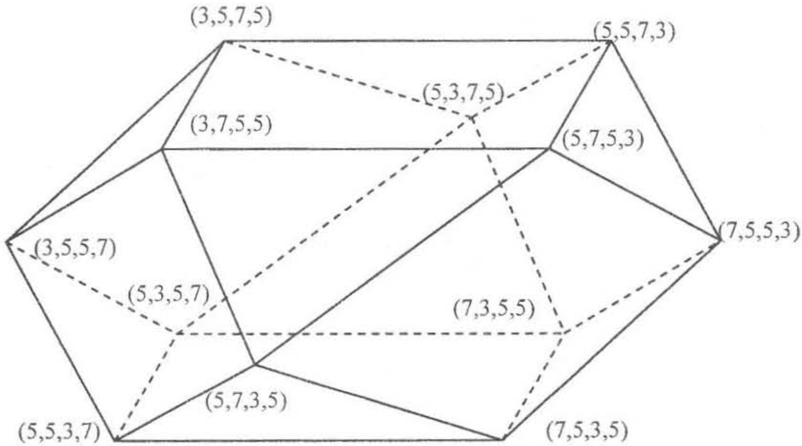

Рис. 2.4

Приклад 2.9. Многогранник $\Pi_{8,6}(G_1)$, де $G_1=\{3,\ 3,\ 4,\ 5,\ 7,\ 9,\ 10,\ 10\}$ за теоремою 2.3 є невиродженим. $[G_1^*]=(1,\ 1,\ 1,\ 1)$. Нехай $G_2=\{5,\ 5,\ 5,\ 8,\ 9,\ 11,\ 14,\ 15\}$. Тоді $[G_2^*]=(1,\ 1,\ 1,\ 1)$. За теоремою 2.8 многогранник $\Pi_{8,6}(G_2)$ також є невиродженим.

2.2. Загальний поліпереставний многогранник

Теорема 2.10. Кількість гіперграней загального многогранника поліпереставлень $\Pi_{kn}^s(G,\ H)$, що збігаються в одній вершині, дорівнює сумі кількості гіперграней, що інцидентні одній вершині кожного з s переставних многогранників $\Pi_{k_i n_i}(G^{N_i})$, тобто:

$$\gamma\left[\Pi_{kn}^s(G,\ H)\right]=\sum_{i=1}^{s}\gamma\left[\Pi_{k_i n_i}\left(G^{N_i}\right)\right].$$

Доведення. Система нерівностей (1.16) складається із сукупності систем, які описують многогранники $\Pi_{k_i n_i}(G^{N_i})$.

Згідно правила, сформульованого в [76], для одержання системи, яка визначає вершину многогранника $\Pi_{kn}(G,\ H)$, необхідно вибрати в незвідній системі кожного з многогранників $\Pi_{k_i n_i}\left(G^{N_i}\right)$ нерівності, які визначають його гіперграні, що збігаються в одній і тій же вершині і записати їх у вигляді рівнянь разом з рівняннями системи (1.16).

Для формування системи гіперграней, що збігаються в одній вершині, візьмемо нерівності, які визначають гіперграні, інцидентні одній вершині многогранника $\Pi_{k_1 n_1}\left(G^{N_1}\right)$ (їх буде $\gamma\left[\Pi_{k_1 n_1}\left(G^{N_1}\right)\right]$); додамо до них відповідні нерівності із системи, що описує $\Pi_{k_2 n_2}\left(G^{N_2}\right)$, кількість яких складає величину $\gamma\left[\Pi_{k_2 n_2}\left(G^{N_2}\right)\right]$, і так далі до многогранника $\Pi_{k_s n_s}\left(G^{N_s}\right)$, додавши до загальної кількості нерівностей величину $\gamma\left[\Pi_{k_s n_s}\left(G^{N_s}\right)\right]$. Тоді, саме ці нерівності, записані у вигляді рівнянь, разом з рівняннями системи (1.16) і будуть складати систему гіперграней $\Pi_{kn}^{s}(G,\ H)$, які збігаються в одній його вершині.

Виходячи з цього, а також з того факту, що кожна гіпергрань многогранника $\Pi_{k_i n_i}\left(G^{N_i}\right)$ $\forall i \in J_s$ породжує гіпергрань многогранника $\Pi_{kn}(G,\ H)$ [76], робимо висновок, що $\gamma\left[\Pi_{kn}^{s}(G,\ H)\right]=\gamma\left[\Pi_{k_1 n_1}\left(G^{N_1}\right)\right]+$ $+\gamma\left[\Pi_{k_2 n_2}\left(G^{N_2}\right)\right]+\ldots+\gamma\left[\Pi_{k_s n_s}\left(G^{N_s}\right)\right]=\sum_{i=1}^{s}\gamma\left[\Pi_{k_i n_i}\left(G^{K_i}\right)\right]$. Теорему доведено.

Теорема 2.11. Многогранник $\Pi_{kn}(G,\ H)$, що описується незвідною системою (1.16) є невиродженим тоді і тільки тоді, якщо:

$$r_i^j=1 \quad \forall i \in J_{n-1}\setminus\{1\} \quad \forall j \in J_s,\ \text{де}\ r_i^j \in \left[G^{N_j}\right]. \quad (2.3)$$

Доведення. Достатність. Нехай має місце (2.3). За теоремою 2.3 кожен з многогранників $\Pi_{k_i n_i}\left(G^{N_i}\right)$ є невиродженим. Звідси $\gamma\left[\Pi_{k_i n_i}\left(G^{N_i}\right)\right]=k_i-1\ \forall i \in J_s$. За теоремою 2.10 $\gamma\left[\Pi_{kn}^{s}(G,\ H)\right]=\sum_{i=1}^{s}\gamma\left[\Pi_{k_i n_i}\left(G^{N_i}\right)\right]=\sum_{i=1}^{s}(k_i-1)=\sum_{i=1}^{s}k_i-s==k-s$.

Відомо (див., напр. [163]), що вимірність многогранника, який є добутком s многогранників M_i визначається так: $dim\left(\bigotimes_{i=1}^{s} M_i\right)=\sum_{i=1}^{s} dim M_i$. Звідси $dim\,\Pi_{kn}^{s}(G,\ H)=\sum_{i=1}^{s} dim\left(\Pi_{k_i n_i}\left(G^{N_i}\right)\right)=$
$=\sum_{i=1}^{s}(k_i-1)=k-s$.

Отже, за твердженням 2.2 достатність доведено.

Необхідність. Нехай $\Pi_{kn}^{s}(G,\ H)$ – невироджений, тоді, з твердження 2.2 випливає, що він є простим і $\gamma\left[\Pi_{kn}^{s}(G,\ H)\right]=k-s$. Оскільки $k-s=\sum_{i=1}^{s}(k_i-1)$, то з теореми 2.10 випливає, що кожна вершина многогранника $\Pi_{k_i n_i}\left(G^{N_i}\right)$ інцидентна k_i-1 гіперграням, тобто кожен з цих многогранників є невиродженим. Таким чином, повинна виконуватися необхідна умова теореми 2.3, що доводить справедливість (2.3). Теорему доведено.

Приклад 2.12. Нехай $G=\{2,5,5,8,9,9\}$. Маємо $k=6$, $n=4$, $r_1=1$, $r_2=2$, $r_3=1$, $r_4=r_n=2$. Розіб'ємо множину $J_k=J_6$ на три підмножини двома способами:

1) $N_1^{(1)}=\{2,3,5,6\}$, $N_2^{(1)}=\{1\}$, $N_3^{(1)}=\{4\}$; $G^{N_1^{(1)}}=\{5,5,9,9\}$, $G^{N_2^{(1)}}=\{2\}$, $G^{N_3^{(1)}}=\{8\}$;

2) $N_1^{(2)}=\{2,3,4,5\}$, $N_2^{(2)}=\{1\}$, $N_3^{(2)}=\{6\}$; $G^{N_1^{(2)}}=\{5,5,8,9\}$, $G^{N_2^{(2)}}=\{2\}$, $G^{N_3^{(2)}}=\{9\}$.

$dim\,\Pi_{6,4}^{3}\left(G,\ H^{(1)}\right)=dim\,\Pi_{6,4}^{3}\left(G, H^{(2)}\right)=3$. За теоремою 2.10 $\gamma\left(\Pi_{6,4}^{3}\left(G,\ H^{(1)}\right)\right)=4$, а $\gamma\left(\Pi_{6,4}^{3}\left(G,\ H^{(2)}\right)\right)=3$. Отже, за теоремою 2.11 $\Pi_{6,4}^{3}\left(G,\ H^{(1)}\right)$ – вироджений (рис. 2.5), а $\Pi_{6,4}^{3}\left(G,\ H^{(2)}\right)$– невироджений поліпереставні многогранники (рис. 2.6).

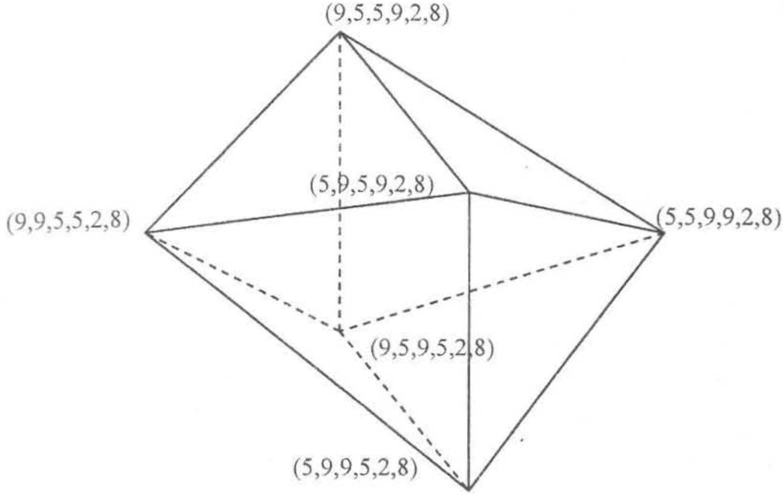
(9,5,5,9,2,8)
(5,9,5,9,2,8)
(9,9,5,5,2,8)
(5,5,9,9,2,8)
(9,5,9,5,2,8)
(5,9,9,5,2,8)

Рис. 2.5

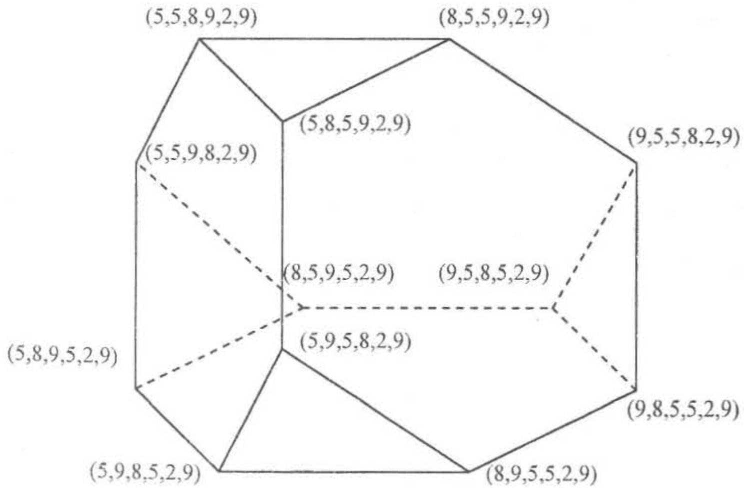
(5,5,8,9,2,9)
(8,5,5,9,2,9)
(5,8,5,9,2,9)
(9,5,5,8,2,9)
(5,5,9,8,2,9)
(8,5,9,5,2,9)
(9,5,8,5,2,9)
(5,9,5,8,2,9)
(5,8,9,5,2,9)
(9,8,5,5,2,9)
(5,9,8,5,2,9)
(8,9,5,5,2,9)

Рис. 2.6

Теорема 2.13 (Наслідок з теореми 2.11). Нехай $[\tilde{G}_j^{N_t}] \subset [G_j^{N_t}]$ і $\tilde{r}_{ij}^t = r_{ij}^t \ \forall \ t \in J_s$, $\forall i \in J_n$, $i \neq 1$, $i \neq n$, де r_{ij}^t, $\tilde{r}_{ij}^t$ – кількості повторень елемента e_i в мультимножинах $G_j^{N_t}$ і $\tilde{G}_j^{N_t}$ відповідно, $j \in J_2$; $n > 2$. Тоді, якщо $\Pi_{kn}^s(G_1, H)$ – невироджений многогранник в R^k, що описується незвідною системою вигляду (1.16), і $[\tilde{G}_1^{N_t}] = [\tilde{G}_2^{N_t}] \ \forall \ t \in J_s$, то $\Pi_{kn}^s(G_2, H)$ також є невиродженим.

Доведення. Оскільки за умовою $\Pi_{kn}^s(G_1, H)$ є невиродженим многогранником, то за теоремою 2.11 для кожної з t первинних специфікацій $[G_1^{N_t}]$ маємо: $r_{i1}^t = 1 \quad \forall i \in J_{n-1} \setminus \{1\}$, $\forall \ t \in J_s$, тобто $[G_1^{N_t}] = (\alpha, \ 1, \ ..., \ 1, \ \beta)$, де α і β – цілі додатні числа. За умовою теореми $[\tilde{G}_1^{N_t}] = [\tilde{G}^{N_t}] \ \forall \ t \in J_s$. Отже, за теоремою 2.11 $\Pi_{kn}^s(G \ , \ H)$ також є невиродженим, що і треба було довести.

2.3. Про еквівалентність опуклих оболонок евклідових комбінаторних множин

Нехай M – многогранник вимірності k в просторі R^k. Множину всіх граней многогранника M вимірностей, що не перевищують $k-1$ називають [38] граничним комплексом многогранника.

Означення 2.14. [38, с. 80] Многогранник M називаємо поміченим многогранником, якщо його граням максимальної вимірності приписані мітки, наприклад, числа 1, 2, ..., $f_{d-1}(M)$. Два помічених многогранника M і M' називаються еквівалентними (позначається $M \sim M'$), якщо існує ізоморфізм їх граничних комплексів, що зберігає мітки граней.

Нехай система обмежень, що задає многогранник M така:

$$A \cdot X \geq b. \tag{2.4}$$

Означення 2.15. Дві евклідові комбінаторні множини E_1 і E_2 назвемо евклідовими комбінаторними множинами одного типу, якщо системи обмежень вигляду (2.4), що описують $conv \ E_1$ і $conv \ E_2$, мають одну й ту ж матрицю A.

Множинами одного типу є загальні множини переставлень, оскільки в просторі R системи (1.4), що описують їхні опуклі оболонки, мають однакові ліві частини обмежень. Те ж саме стосується і загальних множин поліпереставлень із спільною множиною H, тобто з однаковим розбиттям множини J_k на s підмножин. Очевидно, що при цьому у поліпереставних множин одного типу збігаються відповідні ліві частини систем обмежень виду (1.16), що описують відповідні многогранники поліпереставлень.

Розглянемо еквівалентність многогранників при фіксованій матриці A. Позначимо многогранник, що описується (2.4), $M(b)$.

Означення 2.16. [38, с. 81] Спектром $S(b_1,\ b_2)$ простих многогранників $M(b_1)$, $M(b_2)$ при фіксованій матриці A називається множина усіх таких чисел $\lambda \in (0,1)$, що многогранник $M(b_\lambda)$ – вироджений. Тут $b_\lambda = \lambda b + (1-\lambda) b_2$.

Твердження 2.17. [38, с. 81] Два простих многогранника $M(b\ )$ і $M(b_2)$ еквівалентні тоді і тільки тоді, коли їх спектр є порожнім.

Не обмежуючи загальності міркувань, можна вважати, що елементи мультимножини G задовольняють умову (1.2).

Теорема 2.18. Два невироджених многогранника $\Pi(G_1)$ і $\Pi(G\ )$, що є опуклими оболонками евклідових комбінаторних множин переставлень (поліпереставлень) одного типу $E(G_1)$ і $E(G\ )$ у просторі R^k, є еквівалентними, якщо збігаються первинні специфікації мультимножин G_1 і G_2, тобто $[G_1]=[G_2]$.

Доведення. Позначимо вектор правих частин нерівностей (2.4) многогранника $\Pi(G_1)$ через b_1, а для $\Pi(G\ )$ – через b. З твердження 2.2 випливає, що многогранники $\Pi(G_1)$ і $\Pi(G\ )$ є простими. Тоді, за твердженням 2.17 для еквівалентності цих многогранників необхідно довести, що многогранник $\Pi(G_\lambda)$ вигляду (2.4) з вектором $b = b_\lambda$:

$$b_\lambda = \lambda b_1 + (1-\lambda) b\ , \qquad (2.5)$$

де

$$\lambda \in (0,1) \qquad (2.6)$$

є невиродженим.

Спираючись на теореми 2.8 та 2.13, достатньо довести, що перетворення (2.5) визначає вектор правих частин многогранника

$\Pi(G_\lambda)$, який можна розглядати як опуклу оболонку евклідової комбінаторної множини $E(G_\lambda)$, де елементи $g_i^\lambda \in G_\lambda$ визначаються за правилом

$$g_i^\lambda = \lambda g_i^1 + (1-\lambda) g_i^2 \quad \forall i \in J_k \tag{2.7}$$

і при цьому рівність $[G_1]=[G_\lambda]=[G_2]$ виконується для всіх значень λ, на які накладена умова (2.6).

Для доведення рівності первинних специфікацій покажемо, що для будь-яких λ з (2.6) однаковим елементам з G_1 і G_2 відповідають однакові елементи з G_λ, тобто:

$$\begin{cases} g_j^1 = g_l^1, \\ g_j^2 = g_l^2, \end{cases} \Rightarrow g_j^\lambda = g_l^\lambda \quad \forall j,l \in J_k, \tag{2.8}$$

де $g_j^1,\ g_l^1 \in G_1,\ g_j^2,\ g_l^2 \in G_2,\ g_j^\lambda,\ g_l^\lambda \in G_\lambda,\ j \neq l$.

Розглянемо рівність $g_j^\lambda = g_l^\lambda$

$$\begin{aligned} &\lambda g_j^1 + (1-\lambda) g_j^2 = \lambda g_l^1 + (1-\lambda) g_l^2, \\ &\lambda\left(g_j^1 - g_l^1\right) = (1-\lambda)\left(g_l^2 - g_j^2\right) \quad . \end{aligned} \tag{2.9}$$

Оскільки $g_j^1 = g_l^1$ і $g_j^2 = g_l^2$, то $g_j - g_l = g_l^2 - g_j^2 = 0$ і рівність (2.9) виконується для всіх значень λ з умови (2.6).

Покажемо також, що різні елементи мультимножин G_1 і G_2 генерують за правилом (2.7) різні елементи мультимножини G_λ, тобто

$$\begin{cases} g_j^1 \neq g_l^1, \\ g_j^2 \neq g_l^2, \end{cases} \Rightarrow g_j^\lambda \neq g_l^\lambda \quad \forall j,l \in J_k. \tag{2.10}$$

Доведемо (2.10) від супротивного.

Нехай

$$\begin{cases} g_j^1 \neq g_l^1, \\ g_j^2 \neq g_l^2, \end{cases} \Rightarrow g_j^\lambda = g_l^\lambda. \tag{2.11}$$

Тоді маємо:

$$\lambda\left(g_j^1 - g_l^1\right) = (1-\lambda)\left(g_l^2 - g_j^2\right),$$

$$\frac{1-\lambda}{\lambda} = \frac{g_j^1 - g_l^1}{g_l^2 - g_j^2}.$$

Позначимо $\frac{g_j^1 - g_l^1}{g_l^2 - g_j^2} = a$. Нехай $j > l$. Тоді з (1.2) випливає: $g_j^1 > g_l^1$ і $g_j^2 > g_l^2$. Якщо $j < l$, то $g_j < g_l$ і $g_j^2 < g_l^2$.

У будь-якому випадку величина $a < 0$.

$$\frac{1-\lambda}{\lambda} = a, \qquad\qquad (2.12)$$

$$\lambda = \frac{1}{a+1}.$$

Розглянемо три можливих випадки оцінки значень у виразі (2.12):

1) якщо $-1 < a < 0$, то $0 < a+1 < 1 \Rightarrow \lambda > 1$;

2) якщо $a < -1$, то $a+1 < 0 \Rightarrow \lambda < 0$;

3) якщо $a = -1$, то $\frac{1-\lambda}{\lambda} = -1$; 1 = 0, а це неможливо. Отже, такого λ не існує.

Таким чином, жодне із значень λ не належить інтервалу (0; 1). Отже, виконання умови (2.11) суперечить (2.6). Теорему доведено.

РОЗДІЛ 3
ПОЛІКОМБІНАТОРНА ОПТИМІЗАЦІЯ НА МНОЖИНАХ РОЗМІЩЕНЬ

3.1. Незвідна система обмежень для загального многогранника розміщень

Серед різних типів прикладних задач, для моделювання та розв'язування яких використовується апарат евклідової комбінаторної оптимізації, значне місце займають задачі на розміщеннях [40; 45; 50; 51; 62; 63; 67].

Цей розділ присвячений одному з суттєвих етапів дослідження евклідових комбінаторних множин – установленню незвідної системи лінійних обмежень. У даному підрозділі така система встановлена для загального многогранника розміщень – опуклої оболонки загальної множини розміщень.

Поставимо задачу встановити незвідну систему обмежень для загального многогранника розміщень (1.5).

Запишемо систему, що описує $\Pi^k_{\eta n}(G)$, в формі:

$$\begin{cases} \sum_{j=1}^{|\omega|} g_j \le \sum_{i\in\omega} x_i, & (3.1) \\ \sum_{i\in\omega} x_i \le \sum_{j=1}^{|\omega|} g_{\eta-j+1}, \quad \forall \omega \subset J_k & (3.2) \end{cases}$$

і введемо деякі необхідні поняття.

Назвемо $|\omega|$-спілкою сукупність нерівностей системи (3.1), (3.2), які мають однакове значення $|\omega|$.

Для $\Pi^k_{\eta n}(G)$ назвемо характеристичною точкою гіперплощини $\sum_{i\in\omega} x_i = \sum_{j=1}^{|\omega|} g_j$, яка визначається нерівністю (3.1), точку $x' = (x'_1, x'_2, \dots, x'_k)$, якщо

$$x'_{\alpha_1} = x'_{\alpha_2} = \dots = x'_{\alpha_{|\omega|}} = \left(g_1 + g_2 + \dots + g_{|\omega|}\right)/|\omega|,$$

$$x'_{\alpha_{|\omega|+1}} = \dots = x'_{\alpha_k} = \left(g_{|\omega|+1} + \dots + g_k\right)/\left(k - |\omega|\right).$$

Аналогічно, характеристична точка гіперплощини $\sum_{i\in\omega} x_i = \sum_{j=1}^{|\omega|} g_{\eta-j+1}$, яка визначається нерівністю (3.2) – це точка $x' = (x_1', x_2', \dots, x_k')$, координати якої задовольняють співвідношенням:

$$x'_{\alpha_1} = \dots = x'_{\alpha_{|\omega|}} = \left(g_\eta + g_{\eta-1} + \dots + g_{\eta-|\omega|+1}\right)/|\omega|,$$

$$x'_{\alpha_{|\omega|+1}} = \dots = x'_{\alpha_k} = \left(g_1 + \dots + g_{\eta-|\omega|}\right)/\left(\eta-|\omega|\right).$$

Нехай у многограннику $\Pi_{\eta n}^k(G)$ мультимножина G така: $G = \left\{e_1^{\eta_1}, e_2^{\eta_2}, \dots, e_n^{\eta_n}\right\}$.

Теорема 3.1. Якщо в первинній специфікації мультимножини G $\eta_1 > 1$, або $\eta_n > 1$, то надлишковими обмеженнями системи (3.1), (3.2) многогранника $\Pi_{\eta n}(G)$ є:

– серед нерівностей підсистеми (3.1) при $\eta_1 > 1$ – усі нерівності i-спілок, де i задовольняє умовам:

$$2 \le i \le \eta_1; \quad \eta - \eta_n \le i \le k-1; \quad (3.3)$$

– серед нерівностей підсистеми (3.2) при $\eta_n > 1$ – усі нерівності i-спілок, де i таке:

$$2 \le i \le \eta_n; \quad \eta - \eta_1 \le i \le k-1 \qquad (3.4)$$

і тільки вони.

Доведення. Як відомо з [24], якщо для многогранника $\Pi_{\eta n}^k(G)$ маємо $\eta_1 > 1$, або $\eta_n > 1$, то надлишковими обмеженнями в системі (3.1), (3.2) при $\eta_1 > 1$ є нерівності виду (3.1), для яких $1 < |\omega| \le \eta_1$, а при $\eta_n > 1$ нерівності виду (3.2), для яких $1 < |\omega| \le \eta_n$.

Для доведення теореми доведемо спочатку твердження, що сформульовано в [103] у вигляді леми.

Лема 3.2. Якщо $k > \eta - \eta_n$, то надлишковими обмеженнями серед нерівностей (3.1) є нерівності, для яких $\eta - \eta_n \le |\omega| < k$. Якщо

$k > \eta - \eta_1$, то серед нерівностей (3.2) надлишковими є нерівності, для яких $\eta - \eta_1 \le |\omega| < k$.

Доведення леми. Якщо $1 < |\omega| \le \eta_1$, то нерівності (3.1) мають вигляд $\sum_{i\in\omega} x_i \ge |\omega| \cdot g_1$, тобто є лінійними комбінаціями нерівностей $x_i \ge g_1 \quad \forall i \in J_k$. Аналогічно, якщо $1 < |\omega| \le \eta_n$, то нерівності (3.1) мають вигляд $\sum_{i\in\omega} x_i \le |\omega| \cdot g_\eta$, тобто є лінійними комбінаціями нерівностей $x_i \le g_\eta \quad \forall i \in J_k$.

При цьому для $|\omega| = k$ маємо відповідно:

$$\sum_{i\in\omega} x_i \ge \sum_{j=1}^{k} g_j = \sum_{j=1}^{\eta-\eta_n} g_j + \sum_{j=\eta-\eta_n+1}^{k} g_j = \sum_{j=1}^{\eta-\eta_n} g_j + (k - \eta + \eta_n) g_\eta, \quad (3.5)$$

$$\sum_{i\in\omega} x_i \le \sum_{j=1}^{k} g_{\eta-j+1} = \sum_{j=1}^{\eta-\eta_1} g_{\eta-j+1} + \sum_{j=\eta-\eta_1+1}^{k} g_{\eta-j+1} =$$

$$= \sum_{j=1}^{\eta-\eta_1} g_{\eta-j+1} + (k - \eta + \eta_1) g_1, \quad (3.6)$$

Якщо $\eta - \eta_n \le |\omega| < k$, то

$$\sum_{i\in\omega} x_i \ge \sum_{j=1}^{\eta-\eta_n} g_j + \sum_{j=\eta-\eta_n+1}^{|\omega|} g_j = \sum_{j=1}^{\eta-\eta_n-1} g_j + (|\omega| - \eta + \eta_n) g_\eta. \quad (3.7)$$

Зі співвідношення (3.5) випливає, що $\sum_{j=1}^{\eta-\eta_n} g_j = \sum_{j=1}^{k} g_j - (k - \eta + \eta_n) g_\eta$. Підставивши даний вираз у співвідношення (3.7), одержимо нерівність $\sum_{i\in\omega} x_i \ge \sum_{j=1}^{k} g_j - (k - |\omega|) g_\eta$. Отже, при $k > \eta - \eta_n$ нерівності (3.1) є лінійними комбінаціями нерівностей (3.5) та нерівностей $x_i \le g_\eta$, $i \in J_k$, тобто є надлишковими.

Якщо $\eta - \eta_1 \le |\omega| < k$, то

$$\sum_{i\in\omega} x_i \le \sum_{j=1}^{\eta-\eta_1} g_{\eta-j+1} + \sum_{j=\eta-\eta_1+1}^{|\omega|} g_{\eta-j+1} = \sum_{j=1}^{\eta-\eta_1} g_{\eta-j+1} + (|\omega| - \eta + \eta_1) g_1 \quad (3.8)$$

Аналогічно зі співвідношення (3.6) випливає рівність: $\sum_{j=1}^{\eta-\eta_1} g_{\eta-j+1} = \sum_{j=1}^{k} g_{\eta-j+1} - (k-\eta+\eta_1) g_1$. Отже, враховуючи співвідношення (3.8) маємо: $\sum_{i\in\omega} x_i \le \sum_{j=1}^{k} g_{\eta-j+1} + (k-|\omega|)\cdot g_1$, тобто, при $k > \eta - \eta_1$ нерівності (3.2) є лінійними комбінаціями нерівностей (3.6) та нерівностей $x_i \ge g_1 \quad \forall i \in J_k$, а тому вони є надлишковими. Лему доведено.

При $\eta_1 = k$ після вилучення з підсистеми (3.1) нерівностей i-спілок, де i задовольняє (3.3), ця підсистема складатиметься тільки з нерівностей першої спілки, які, очевидно, не є надлишковими. Аналогічно, у випадку $\eta_n = k$, після вилучення з підсистеми (3.2) надлишкових нерівностей i-спілок, де i задовольняє (3.4), у ній залишаться лише ненадлишкові нерівності першої спілки цієї підсистеми. Отже, залишається довести, що при $1 < \eta_1 < k$ підсистема (3.1) не має надлишкових нерівностей, відмінних від нерівностей i-спілок, де i задовольняє (3.3), а при $1 < \eta_n < k$ підсистема (3.2) не має надлишкових нерівностей, відмінних від нерівностей i-спілок, де i задовольняє (3.4).

Доведемо незвідність підсистеми (3.1) за наступним планом.

I. Встановимо, що в характеристичній точці гіперплощини, яка визначається довільно вибраною нерівністю будь-якої фіксованої спілки m, де m не задовольняє (3.3), кожне з обмежень підсистеми (3.1), за винятком вибраної нерівності, перетворюється в строгу числову нерівність. Для цього розглядаються нерівності таких спілок.

I (А). Нерівності спілок m, для яких виконується умова

$$\eta_1 < m < min\{k, \eta - \eta_n\}. \qquad (3.9)$$

I (Б). Нерівності спілки 1.

I (В). Нерівність спілки k.

II. Доводиться, що виключення з підсистеми (3.1) нерівності з будь-якої i-спілки, відмінної від i-спілок, де i задовольняє (3.3), приводить до зміни многогранника $\Pi_{\eta n}^{k}(G)$.

Доведення незвідності підсистеми (3.1).

I (А). Виберемо довільну нерівність

$$x'_{\alpha_1} + x'_{\alpha_2} + \ldots + x'_{\alpha_m} \ge g_1 + g_2 + \ldots + g_m \qquad (3.10)$$

будь-якої фіксованої спілки m, де m задовольняє умову (3.9). Нехай точка $x^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, \ldots, x_k^{(0)}\right)$ – характеристична точка гіперплощини, яка визначається нерівністю (3.10). Будемо вважати, що

$$\left(g_1 + g_2 + \ldots + g_m\right) / m = \alpha, \ \left(g_{m+1} + g_{m+2} + \ldots + g_k\right) / (k - m) = \beta,$$

тобто $x_1^{(0)} = x_2^{(0)} = \ldots = x_m^{(0)} = \alpha, x_{m+1}^{(0)} = x_{m+2}^{(0)} = \ldots = x_k^{(0)} = \beta$.

Оскільки елементи мультимножини G упорядковані за неспаданням і виконуються умови (3.9), (3.10), то

$$g_1 < \alpha < g_m, \tag{3.11}$$

$$g_{m+1} < \beta < g_k, \tag{3.12}$$

$$\alpha < \beta. \tag{3.13}$$

Отже, існують такі значення t_1 і t_2, для яких виконуються співвідношення:

$$g_1 \le g_2 \le \ldots \le g_{t_1} < \alpha \le g_{t_1+1} \le g_{t_1+2} \le \ldots \le g_k, \tag{3.14}$$

$$g_1 \le g_2 \le \ldots \le g_{t_2} < \beta \le g_{t_2+1} \le g_{t_2+2} \le \ldots \le g_k, \tag{3.15}$$

причому $1 < t_1 < m, \ m + 1 \le t_2 < k$. У точці $x^{(0)}$ нерівність (3.10) перетворюється на числову рівність

$$m\alpha = g_1 + g_2 + \ldots + g_m. \tag{3.16}$$

Покажемо, що всі нерівності $|\omega|$-спілок, $|\omega| \in J_{k-1} \setminus \{1\}$, підсистеми (3.1) у точці x перетворюються в строгі. Для цього розглянемо три випадки: 1) $|\omega| = m$, 2) $1 < |\omega| < m$, 3) $m < |\omega| < k$.

1. Будь-яка нерівність спілки $|\omega| = m$, відмінна від нерівності (3.10), містить u, $(u \in J_m)$, змінних x_j, де $j \in J_k \setminus J_m$, а тому в точці x її ліва частина дорівнює $u\beta + (m - u)\alpha$. Оскільки виконується нерівність (3.13), то

$$u\beta + (m - u)\alpha = u(\beta - \alpha) + m\alpha > g_1 + g_2 + \ldots + g_m.$$

Отже, кожне з обмежень спілки m, за виключенням обмеження (3.10), в точці x перетворюється в строгу нерівність.

2. Розглянемо спочатку нерівності спілок $|\omega|$, $1 < |\omega| < m$, ліві частини яких містять тільки змінні x , де $j \in J_m$. У точці x кожна з них набуває вигляду $|\omega| \alpha \ge g_1 + g_2 + \ldots + g_{|\omega|}$. Якщо $0 < |\omega| \le t_1$, то, врахо-

вуючи співвідношення (3.14), маємо: $|\omega|\alpha > g_1 + g_2 + ... + g_{|\omega|}$. За умови $t_1 < |\omega| < m$ із співвідношень (3.11) і (3.14) випливає нерівність

$$\left(m - |\omega|\right)\alpha < g_{|\omega|+1} + g_{|\omega|+2} + ... + g_m. \qquad (3.17)$$

Дійсно, ліва її частина містить $m-|\omega|$ доданків α, де $\alpha \le g_j$, а права містить $m-|\omega|$ доданків g_j, серед яких $g_m > \alpha$, $j \in J_m \setminus J_{|\omega|}$. Розглянемо рівність

$$g_1 + g_2 + ... + g_{|\omega|} = g_1 + g_2 + ... + g_m - \left(g_{|\omega|+1} + g_{|\omega|+2} + ... + g_m\right).$$

Оскільки має місце співвідношення (3.16), то

$$g_1 + g_2 + ... + g_{|\omega|} = m\alpha - \left(g_{|\omega|+1} + g_{|\omega|+2} + ... + g_m\right).$$

Враховуючи нерівність (3.17), одержимо строгу нерівність:

$|\omega|\alpha = m\alpha - \left(m - |\omega|\right)\alpha > m\alpha - \left(g_{|\omega|+1} + g_{|\omega|+2} + ... + g_m\right) = g_1 + g_2 + ... + g_{|\omega|}$.

Отже, всі нерівності спілок $|\omega|$, $1<|\omega|<m$, які містять тільки змінні x_j, де $j \in J_m$, в точці $x^{(0)}$ перетворюються в строгі.

Виберемо тепер будь-яку іншу нерівність спілки $|\omega|$, $1<|\omega|<m$, тобто таку, яка містить v, $\left(v \in J_{|\omega|}\right)$, змінних x, де $j \in J_k \setminus J_m$. Ліва частина такої нерівності в точці $x^{(0)}$ набуває вигляду: $v\beta + \left(|\omega| - v\right)\alpha = v\left(\beta - \alpha\right) + |\omega|\alpha$. Враховуючи нерівність (3.13), маємо:

$$v\beta + \left(|\omega| - v\right)\alpha = v\left(\beta - \alpha\right) + |\omega|\alpha > |\omega|\alpha \ge g_1 + g_2 + ... + g_{|\omega|}.$$

Таким чином, усі нерівності спілок $|\omega|$, $1<|\omega|<m$, у точці $x^{(0)}$ перетворюються в строгі.

3. Аналогічно до випадку 2), спочатку розглянемо тільки нерівності підсистеми (3.1) спілок $|\omega|$, $m< |\omega|<k$, які містять усі змінні x, де $j \in J_m$. У точці $x^{(0)}$ кожна з цих нерівностей перетворюється в таку:

$$m\alpha + \left(|\omega| - m\right)\beta \ge g_1 + g_2 + ... + g_m + g_{m+1} + ... + g_{|\omega|}.$$

Якщо $m<|\omega|<t_2$ $(m<t_2<k)$, то із співвідношення (3.15) випливає нерівність $(|\omega|-m)\beta>g_{m+1}+g_{m+2}+...+g_{|\omega|}$ (ліва її частина містить $|\omega|-m$ доданків β, $(\beta>g_j)$, а права частина– $|\omega|-m$ доданків $g_j, j\in J_k\setminus J_{|\omega|}$). Беручи до уваги рівність (3.16), маємо:

$$m\alpha+(|\omega|-m)\beta>g_1+g_2+...+g_m+g_{m+1}+...+g_{|\omega|}.$$

Нехай $t_2<|\omega|<k$. За цієї умови, в силу співвідношень (3.12), (3.15), має місце нерівність

$$(k-|\omega|)\beta<g_{|\omega|+1}+g_{|\omega|+2}+...+g_k. \qquad (3.18)$$

Розглянемо рівність

$$g_1+g_2+...+g_{|\omega|}=g_1+g_2+...+g_k-\left(g_{|\omega|+1}+g_{|\omega|+2}+...+g_k\right).$$

Оскільки $m\alpha+(k-m)\beta=g_1+g_2+...+g_k$, то

$$g_1+...+g_{|\omega|}=m\alpha+(k-m)\beta-\left(g_{|\omega|+1}+g_{|\omega|+2}+...+g_k\right).$$

Враховуючи нерівність (3.18), маємо:

$$m\alpha+(|\omega|-m)\beta=m\alpha+(k-m)\beta-(k-|\omega|)\beta>m\alpha+$$

$$+(k-m)\beta-\left(g_{|\omega|+1}+g_{|\omega|+2}+...+g_k\right)=g_1+g_2+...+g_{|\omega|}.$$

Отже, нерівності спілок $|\omega|$, $m<|\omega|<k$, які містять усі змінні x_j, де $j\in J_m$, у точці $x^{(0)}$ перетворюються в строгі.

Тепер розглянемо нерівності довільної спілки $|\omega|, m<|\omega|<k$, які містять не всі змінні x, де $j\in J_m$. Будь-яка з таких нерівностей містить на μ змінних x більше, (де $j\in J_k\setminus J_m, 1\le\mu\le m$), порівняно із уже розглянутими нерівностями цієї ж спілки. Отже, ліва її частина в точці $x^{(0)}$ дорівнює $(m-\mu)\alpha+(|\omega|-m+\mu)\beta$. Беручи до уваги умову (3.13), маємо:

$$(m-\mu)\alpha+(|\omega|-m+\mu)\beta=m\alpha+(|\omega|-m)\beta+\mu(\beta-\alpha)>$$

$$>m\alpha+(|\omega|-m)\beta>g_1+g_2+\ldots+g_{|\omega|}.$$

Отже, у точці $x^{(0)}$ усі нерівності спілок $|\omega|$, де $m<|\omega|<k$, перетворюються в строгі.

Таким чином, розглянуто всі з трьох можливих випадків і доведено, що будь-яка нерівність довільної $|\omega|$-спілки, де $|\omega|\in J_{k-1}\setminus\{1\}$, за виключенням нерівності (3.10), в характеристичній точці x гіперплощини, що визначається нерівністю (3.10) за умови (3.9), перетворюється в строгу нерівність. Пункт І (А) доведено.

Сказане вище має місце і у випадках, коли $m=1$ та $m=k$. Доведення аналогічно пункту І (А). Розглянемо його.

І (Б). Нехай $m=1$. Виберемо у цій спілці довільну нерівність

$$x_\omega\ge g_1,\quad \omega\in J_k,\quad |\omega|=1. \tag{3.19}$$

Нехай $x^{(1)}=\left(x_1^{(1)},x_2^{(1)},\ldots,x_k^{(1)}\right)$ – характеристична точка гіперплощини, яка визначається нерівністю (3.19). Позначимо $(g_2+g_3+\ldots+g_k)/(k-1)=\beta^{(1)}$. Тоді $x_1^{(1)}=g_1, x_2^{(1)}=x_3^{(1)}=\ \ldots=x_k^{(1)}=\beta^{(1)}$. Оскільки виконується (1.2), то

$$g_1\le g_2<\beta^{(1)}<g_k. \tag{3.20}$$

Отже, існує таке t_3, для якого має місце нерівність

$$g_1\le g_2\le\ldots\le g_{t_3}<\beta^{(1)}\le g_{t_3+1}\le g_{t_3+2}\le\ldots\le g_k, \tag{3.21}$$

причому $2\le t_3<k$.

Покажемо, що у точці $x^{(1)}$ усі нерівності, за виключенням нерівності (3.19), будь-якої $|\omega|$-спілки, $|\omega|\in J_k$ перетворюються в строгі. Для цього розглянемо такі три випадки: 1) $|\omega|=1$, 2) $1<|\omega|\le t_3$, 3) $t_3<|\omega|<k$.

1. Кожна з нерівностей спілки $|\omega|=1$, за виключенням нерівності (3.19), у точці x перетворюється в строгу числову нерівність $\beta^{(1)}>g_1$.

2. Нехай $1<|\omega|\le t_3$. Розглянемо спочатку ті нерівності, які містять x_ω. У точці $x^{(1)}$ кожна з них набуває вигляду: $g_1+\left(|\omega|-1\right)\beta^{(1)}\ge g_1+g_2+\ldots+g_{|\omega|}$, або $\left(|\omega|-1\right)\beta^{(1)}\ge g_2+\ldots+g_{|\omega|}$. Враховуючи співвідношення (3.21), маємо $\left(|\omega|-1\right)\beta^{(1)}>g_2+\ldots+g_{|\omega|}$. Будь-яка нерівність (зі спілок, що розглядаються), яка не містить x_ω, в силу умови (3.20), у точці $x^{(1)}$ перетворюється в строгу нерівність $|\omega|\beta^{(1)}>g_1+g_2+\ldots+g_{|\omega|}$.

3. Нехай $t_3<|\omega|<k$. Аналогічно до випадку 2 розглянемо спочатку ті нерівності, які містять x_ω. У точці $x^{(1)}$ кожна з них набуває вигляду: $g_1+\left(|\omega|-1\right)\beta^{(1)}>g_1+g_2+\ldots+g_{|\omega|}$. За умови $t_3<|\omega|<k$ із співвідношень (3.20), (3.21) випливає нерівність $\left(k-|\omega|\right)\beta^{(1)}>g_{|\omega|+1}+g_{|\omega|+2}+\ldots+g_k$.

Оскільки $g_1+\left(k-1\right)\beta^{(1)}=g_1+g_2+\ldots+g_k$, то:

$$g_1+\left(|\omega|-1\right)\beta^{(1)}=g_1+\left(k-1\right)\beta^{(1)}-\left(k-|\omega|\right)\beta^{(1)}>g_1+g_2+\ldots+g_k-$$
$$-\left(g_{|\omega|+1}+g_{|\omega|+2}+\ldots+g_k\right)=g_1+g_2+\ldots+g_{|\omega|}.$$

За умови (3.20), кожна з нерівностей даних спілок, яка не містить x_ω, у точці $x^{(1)}$ перетворюється в строгу нерівність $|\omega|\beta^{(1)}>g_1+g_2+\ldots+g_{|\omega|}$.

Таким чином, у характеристичній точці $x^{(1)}$ гіперплощини, яка визначається нерівністю (3.19) 1-спілки, усі нерівності підсистеми (3.1), за виключенням нерівності (3.19), перетворюються в строгі.

І (В). Розглянемо випадок, коли $m=k$. Для нерівності

$$x_1+x_2+\ldots x_k\ge g_1+g_2+\ldots+g_k \qquad (3.22)$$

Позначимо $x^{(2)}=\left(x_1^{(2)},x_2^{(2)},\ldots,x_k^{(2)}\right)$ – характеристичну точку гіперплощини, яка визначається цією нерівністю. Нехай $\left(g_1+g_2+\ldots+g_k\right)/k=\alpha^{(1)}$. Тоді $x_1^{(2)}=x_2^{(2)}=\ldots=x_k^{(2)}=\alpha^{(1)}$. Оскільки виконується умова $1<\eta_1<k$, то

$$g_1<\alpha^{(1)}<g_k. \qquad (3.23)$$

Отже, існує таке t_4, для якого має місце нерівність

$$g_1\le\ldots\le g_{t_4}<\alpha^{(1)}\le g_{t_4+1}\le g_{t_4+2}\le\ldots\le g_k, \qquad (3.24)$$

причому, $1<t_4<k$.

Покажемо, що в точці $x^{(2)}$ усі нерівності, за виключенням нерівності (3.22), будь-якої $|\omega|$-спілки, $|\omega| \in J_k$, перетворюються в строгі. Для цього, як і раніше, розглянемо по черзі такі випадки: 1) $1 < |\omega| \le t_4$, 2) $t_4 < |\omega| < k$, 3) $|\omega| = k$.

1. Нехай $1 < |\omega| \le t_4$. При цій умові із співвідношення (3.24) безпосередньо випливає, що всі нерівності перетворюються в точці x в строгі числові нерівності $|\omega|\alpha^{(1)} > g_1 + g_2 + \ldots + g_{|\omega|}$.

2. Якщо $t_4 < |\omega| < k$, то нерівності в точці x набувають вигляду: $|\omega|\alpha^{(1)} \ge g_1 + g_2 + \ldots + g_{|\omega|}$. За умови $t_4 < |\omega| < k$ із співвідношень (3.23), (3.24) випливає строга нерівність $\left(k - |\omega|\right)\alpha^{(1)} < g_{|\omega|+1} + g_{|\omega|+2} + \ldots + g_k$. Оскільки $k\alpha^{(1)} = g_1 + g_2 + \ldots + g_k$, то

$$|\omega|\alpha^{(1)} = k\alpha^{(1)} - \left(k - |\omega|\right)\alpha^{(1)} > k\alpha^{(1)} - \left(g_{|\omega|+1} + g_{|\omega|+2} + \ldots + g_k\right) =$$

$$= g_1 + g_2 + \ldots + g_k - \left(g_{|\omega|+1} + g_{|\omega|+2} + \ldots + g_k\right) = g_1 + g_2 + \ldots + g_{|\omega|}.$$

3. У спілці $|\omega| = k$ обмеження (3.22) у точці $x^{(2)}$ перетворюється у рівність $k\alpha^{(1)} = g_1 + g_2 + \ldots + g_k$. Ліва частина будь-якої іншої нерівності цієї спілки в точці $x^{(2)}$ набуває вигляду $\left(k-1\right)\alpha^{(1)}$. Оскільки виконується умова (3.23), то $\left(k-1\right)\alpha^{(1)} > k\alpha^{(1)} = g_1 + g_2 + \ldots + g_k$.

Отже, в характеристичній точці $x^{(2)}$ гіперплощини, яка визначається нерівністю (3.22) m-спілки, ($m = k$), усі нерівності, за виключенням (3.22), будь-якої $|\omega|$-спілки, ($|\omega| \in J_k$), перетворюються в строгі нерівності.

II. Залишилося довести, що існує така точка, яка задовольняє усі обмеження підсистеми (3.1), крім вибраної нерівності, тобто виключення з підсистеми (3.1) вибраної нерівності приводить до зміни многогранника $\Pi_{\eta n}^{k}(G)$.

Виберемо з (3.1) довільну нерівність

$$x_{\alpha_1} + x_{\alpha_2} + \ldots + x_{\alpha_m} \ge g_1 + g_2 + \ldots + g_m \quad (3.25)$$

довільної m-спілки, $m \in J_k$, де m не задовольняє (3.3), підсистеми (3.1). Нехай $x^* = (x_1^*, x_2^*, \ldots, x_k^*)$ – характеристична точка гіперплощини, яка визначається нерівністю (3.25). У точці x^* усі нерівності $|\omega|$-спілок, $|\omega| \in J$, за виключенням нерівності (3.25), перетворюються в строгі числові нерівності, що доведено вище (пункт I). Нехай p-кількість таких нерівностей. (Оскільки підсистема (3.1) містить $2^k - 1$ обмежень, то $p = 2^k - 2$). Пронумеруємо ці строгі числові нерівності та позначимо $X_s^* > G_s^*$ ту з них, номер якої s. Позначимо $(g_1 + g_2 + \ldots + g_m)/m = \alpha^*$, $(g_{m+1} + g_{m+2} + \ldots + g_k)/(k-m) = \beta^*$. Тоді $x_1^* = x_2^* = \ldots = x_k^* = \alpha^*$, $x_{m+1}^* = x_{m+2}^* = \ldots = x_k^* = \beta^*$. Зрозуміло, що $\forall s \in J_p \quad \exists \varepsilon(s) > 0: \ X_s^* - \varepsilon(s) > G_s^*$. Нехай $\varepsilon = \min_{s \in J_p} \{\varepsilon(s)\}$.

Розглянемо підсистему (3.1) у точці $\bar{x} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_k)$, де $\bar{x}_j = x_j^* - \varepsilon / m, \ \forall j \in J_m$, тобто $\bar{x}_j = \alpha^* - \varepsilon / m, \ \forall j \in J_m$. У цій точці нерівність (3.25) не виконується, оскільки:

$$\bar{x}_1 + \bar{x}_2 + \ldots + \bar{x}_m = m(\alpha^* - \varepsilon / m) = m\alpha^* - \varepsilon < m\alpha^* = g_1 + g_2 + \ldots + g_m .$$

Усі інші обмеження підсистеми (3.1) у точці $\bar{x}$, в силу вибору ε, виконуються як строгі нерівності. Отже, точка $\bar{x}$ задовольняє всі обмеження підсистеми (3.1), крім нерівності (3.25), а тому виключення з підсистеми (3.1) цієї нерівності приводить до зміни многогранника $\Pi_{\eta n}^k(G)$, тобто, нерівність (3.25) не є надлишковим обмеженням.

З довільності вибору m-спілки, де m не задовольняє (3.3), та довільності вибору у цій спілці нерівності (3.25) випливає справедливість теореми для підсистеми (3.1).

Доведення незвідності підсистеми (3.2). Спочатку розглянемо нерівності підсистеми (3.2) спілок m, для яких виконується умова

$$\eta_n < m < \min\{\eta - \eta_1, \ k\}, \tag{3.26}$$

а потім нерівності спілок 1 та k.

Виберемо довільну нерівність:

$$x'_{\alpha_1} + x'_{\alpha_2} + \ldots + x'_{\alpha_m} \le g_\eta + g_{\eta-1} + \ldots + g_{\eta-m+1} \tag{3.27}$$

будь-якої фіксованої спілки m, якщо m задовольняє умову (3.26). Нехай точка $x^{(0)}=\left(x_1^{(0)},x_2^{(0)},\dots,x_k^{(0)}\right)$ – характеристична точка $(k-2)$ – площини, яка визначається (3.27). Будемо вважати, що

$$\left(g_\eta+g_{\eta-1}+\dots+g_{\eta-m+1}\right)/m=\alpha,\left(g_1+g_2+\dots+g_{\eta-m}\right)/(\eta-m)=\beta,$$

тобто $x_1^{(0)}=x_2^{(0)}=\dots=x_m^{(0)}=\alpha,\ x_{m+1}^{(0)}=x_{m+2}^{(0)}=\dots=x_k^{(0)}=\beta$.

Оскільки елементи мультимножини $g_1\le g_2\le\dots\le g_k$ і виконуються умови (3.26), (3.27), то

$$g_{\eta-m+1}<\alpha<g_\eta, \tag{3.28}$$

$$g_1<\beta<g_{\eta-m}, \tag{3.29}$$

$$\alpha>\beta. \tag{3.30}$$

Отже, існують такі значення t_1 і t_2, для яких виконуються співвідношення (3.14), (3.15), причому $1<t_1<m, m+1\le t_2<k$. У точці $x^{(0)}$ нерівність (3.27) перетворюється на числову рівність

$$m\alpha=g_\eta+g_{\eta-1}+\dots+g_{\eta-m+1}. \tag{3.31}$$

Покажемо, що усі нерівності спілок $|\omega|$, $|\omega|\in J_{k-1}\setminus\{1\}$ підсистеми (3.2) у точці x перетворюються у строгі. Для цього розглянемо випадки: 1) $|\omega|=m$, 2) $1<|\omega|<m$, 3) $m<|\omega|<k$.

Будь-яка нерівність спілки $|\omega|=m$, відмінна від нерівності (3.27) містить $u,\ u\in J_m$, змінних x_j, де $j\in J_k\setminus J_m$, а тому в точці x її ліва частина дорівнює $u\beta+(m-u)\alpha$. Оскільки виконується нерівність (3.30), то $u\beta+(m-u)\alpha=u(\beta-\alpha)+m\alpha<g_\eta+g_{\eta-1}+\dots+g_{\eta-m+1}$. Отже, кожне з обмежень спілки m, за виключенням обмеження (3.29), в точці x перетворюється в строгу нерівність.

2. Розглянемо спочатку нерівності спілок $|\omega|$, $1<|\omega|<m$, ліві частини яких містять тільки змінні x_j, де $j\in J_m$. У точці x кожна з них набуває вигляду: $|\omega|\alpha\le g_\eta+g_{\eta-1}+\dots+g_{\eta-|\omega|+1}$. Якщо $0<|\omega|\le t_1$, то, враховуючи співвідношення (3.14), маємо:

$$|\omega|\alpha<g_\eta+g_{\eta-1}+\dots+g_{\eta-|\omega|+1}.$$

Далі візьмемо $t_1 < |\omega| < m$. При цій умові з співвідношення (3.28) і (3.14) випливає нерівність:

$$\left(m - |\omega|\right)\alpha > g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + \ldots + g_{\eta-m+1}. \quad (3.32)$$

Дійсно, ліва її частина містить $m-|\omega|$ доданків $\alpha \geq g_j$, а права – $m-|\omega|$ доданків g_j, серед яких $g_{\eta-m} < \alpha$, $j \in J_{\eta-m} \setminus J_{|\omega|}$. Розглянемо рівність:

$$g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1} = g_\eta + g_{\eta-1} + \ldots + g_{\eta-m+1} - \left(g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + \ldots + g_{\eta-m+1}\right).$$

Оскільки має місце співвідношення (3.31), то:

$$g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1} = m\alpha - \left(g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + \ldots + g_{\eta-m+1}\right).$$

Враховуючи нерівність (3.32), одержимо строгу нерівність:

$$|\omega|\alpha = m\alpha - \left(m - |\omega|\right)\alpha < m\alpha - \left(g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + \ldots + g_{\eta-m+1}\right) =$$
$$= g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}.$$

Отже, всі нерівності спілок $|\omega|$, $1<|\omega|<m$, які містять тільки змінні x_j, де $j \in J_m$, в точці $x^{(0)}$ перетворюються в строгі.

Виберемо тепер будь-яку іншу нерівність спілки $|\omega|$, $1<|\omega|<m$, тобто таку, яка містить v, $v \in J_{|\omega|}$, змінних x, де $j \in J_k \setminus J_m$. Ліва частина такої нерівності в точці $x^{(0)}$ набуває вигляду: $v\beta + \left(|\omega| - v\right)\alpha = v(\beta - \alpha) + |\omega|\alpha$. Враховуючи нерівність (3.30), маємо:

$$v\beta + \left(|\omega| - v\right)\alpha = v(\beta - \alpha) + |\omega|\alpha < |\omega|\alpha \leq g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}.$$

Таким чином, усі нерівності спілок $|\omega|$, $1<|\omega|<m$, у точці $x^{(\ )}$ перетворюються в строгі.

3. Аналогічно до випадку 2, спочатку розглянемо тільки нерівності спілок $|\omega|$, $m < |\omega| < k$, які містять усі змінні x, де $j \in J_m$. У точці $x^{(\ )}$ кожна з цих нерівностей перетворюється в таку:

$$m\alpha + \left(|\omega| - m\right)\beta \leq g_\eta + g_{\eta-1} + \ldots + g_{\eta-m+1} + g_{\eta-m} + \ldots + g_{\eta-|\omega|+1}.$$

Якщо $m<|\omega|\le t_2$ $(m<t_2<k)$, то із співвідношення (3.15) випливає нерівність $(|\omega|-m)\beta<g_{\eta-m}+g_{\eta-m-1}+\ldots+g_{\eta-|\omega|+1}$ (ліва її частина містить $|\omega|-m$ доданків $\beta<g_j$, а права – $|\omega|-m$ доданків $g_j, j\in J_k\setminus J_{|\omega|}$). Беручи до уваги рівність (3.31), маємо:

$$m\alpha+(|\omega|-m)\beta<g_\eta+g_{\eta-1}+\ldots+g_{\eta-\ +1}+g_{\eta-}\ +\ldots+g_{\eta-|\omega|+1}.$$

Нехай далі $t_2<|\omega|<k$. При цій умові, в силу співвідношень (3.29), (3.15), має місце нерівність:

$$(k-|\omega|)\beta<g_{\eta-|\omega|}+g_{\eta-|\omega|-1}+\ldots+g_{\eta-k+1}. \qquad (3.33)$$

Дійсно, ліва її частина містить $k-|\omega|$ доданків $\beta\ge g_j$, а права – $k-|\omega|$ доданків g_j, $j\in J_k\setminus J_{|\omega|}$, серед яких $g_{\eta-k+1}<\beta$, отже, ця нерівність виконується. Розглянемо рівність:

$$g_\eta+g_{\eta-1}+\ldots+g_{\eta-|\omega|+1}=g_\eta+g_{\eta-1}+\ldots+g_{\eta-k+1}-\left(g_{\eta-|\omega|}+g_{\eta-|\omega|-1}+\ldots+g_{\eta-k+1}\right)$$

Оскільки $m\alpha+(k-m)\beta=g_\eta+g_{\eta-1}+\ldots+g_{\eta-k+1}$, то:

$$g_\eta+g_{\eta-1}+\ldots+g_{\eta-|\omega|+1}=m\alpha+(k-m)\beta-\left(g_{\eta-|\omega|}+g_{\eta-|\omega|-1}+\ldots+g_{\eta-k+1}\right).$$

Враховуючи нерівність (3.33), маємо:

$$m\alpha+(|\omega|-m)\beta=m\alpha+(k-m)\beta-(k-|\omega|)\beta<m\alpha+$$

$$+(k-m)\beta-\left(g_{\eta-|\omega|}+g_{\eta-|\omega|-1}+\ldots+g_{\eta-k+1}\right)=g_\eta+g_{\eta-1}+\ldots+g_{\eta-|\omega|+1}.$$

Отже, нерівності спілок $|\omega|$, $m<|\omega|<k$, які містять усі змінні x_j, де $j\in J_m$, у точці $x^{(0)}$ перетворюються в строгі.

Тепер розглянемо нерівності довільної спілки $|\omega|$, $m<|\omega|<k$, які містять не всі змінні x , де $j\in J_m$. Будь-яка з таких нерівностей містить на μ змінних x , де $j\in J_k\setminus J_m$, $1\le\mu\le m$, більше, порівняно із вже розглянутими нерівностями цієї ж спілки. Отже, ліва її частина

в точці $x^{(0)}$ дорівнює $(m-\mu)\alpha+(|\omega|-m+\mu)\beta$. Беручи до уваги умову (3.30), маємо:

$$(m-\mu)\alpha+(|\omega|-m+\mu)\beta=m\alpha+(|\omega|-m)\beta+\mu(\beta-\alpha)<$$

$$<m\alpha+(|\omega|-m)\beta<g_\eta+g_{\eta-1}+\ldots+g_{\eta-|\omega|+1}.$$

Отже, у точці $x^{(\ )}$ усі нерівності спілок $|\omega|$, де $m<|\omega|<k$, перетворюються в строгі.

Таким чином, розглянуто всі можливі випадки (1), 2), 3)) і отримано, що будь-яка нерівність довільної спілки $|\omega|$, де $|\omega|\in J_{k-1}\setminus\{1\}$, за виключенням нерівності (3.27), в характеристичній точці x $(k-2)$ – площини, що визначається нерівністю (3.27) при умові (3.26), перетворюється в строгу.

Сказане вище має місце і у випадках, коли $m=1$ та $m=k$. Для доведення необхідно зробити так, як і для m, що задовольняє умову (3.26).

Дійсно, нехай $m=1$. Виберемо у цій спілці довільну нерівність:

$$x_\omega\le g_{\eta,}\quad \omega\in J_k,\quad |\omega|=1, \qquad (3.34)$$

Нехай $x^{(1)}=\left(x_1^{(1)},x_2^{(1)},\ldots,x_k^{(1)}\right)$ – характеристична точка $(k-2)$ – площини, яка визначається нерівністю (3.34). Позначимо $(g_{\eta-1}+\ldots+g_{\eta-k+1})/(k-1)=\beta^{(1)}$. Тоді $x_1^{(1)}=g_\eta$, $x_2^{(1)}=x_3^{(1)}=\ldots=x_k^{(1)}=\beta^{(1)}$. Оскільки елементи мультимножини $g_1\le g_2\le\ldots\le g_\eta$, то

$$g_1\le g_2<\beta<g_\eta. \qquad (3.35)$$

Отже, існує таке t_3, для якого має місце співвідношення:

$$g_1\le g_2\le\ldots\le g_{t_3}<\beta^{(1)}\le g_{t_3+1}\le g_{t_3+2}\le\ldots\le g_\eta, \qquad (3.36)$$

причому $2\le t_3<k$.

Покажемо, що у точці $x^{(1)}$ усі нерівності, за виключенням нерівності (3.34), будь-якої спілки $|\omega|\in J_k$ перетворюються в строгі. Для цього розглянемо такі три випадки: 1) $|\omega|=1$, 2) $1<|\omega|\le t_3$, 3) $t_3<|\omega|<k$.

1. Кожна з нерівностей спілки $|\omega|=1$, за виключенням нерівності (3.34), у точці x перетворюється у строгу числову нерівність $\beta^{(1)} < g_\eta$.

2. Нехай $1<|\omega|\le t_3$. Розглянемо спочатку ті нерівності, які містять x_ω. У точці x кожна з них набуває вигляду: $g_\eta + (|\omega|-1)\beta \le g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}$. Віднявши від обох частин цієї нерівності g_η, одержимо: $(|\omega|-1)\beta^{(1)} \le g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}$. Враховуючи співвідношення (3.36), маємо $(|\omega|-1)\beta^{(1)} < g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}$. Будь-яка нерівність (спілок, що розглядаються), яка не містить x_ω, в силу умови (3.35), у точці $x^{(1)}$ перетворюється в строгу нерівність:

$$|\omega|\beta^{(1)} < g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}.$$

3. Нехай $t_3 < |\omega| < k$. Аналогічно до випадку 2 розглянемо спочатку ті нерівності, які містить x_ω. У точці $x^{(1)}$ кожна з них набуває вигляду $g_\eta + (|\omega|-1)\beta^{(1)} > g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}$. При умові $t_3 < |\omega| < k$ із співвідношень (3.35), (3.36) випливає нерівність:

$$(k-|\omega|)\beta^{(1)} > g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + \ldots + g_{\eta-k+1}.$$

Отже, оскільки $g_\eta + (k-1)\beta^{(1)} = g_\eta + g_{\eta-1} + \ldots + g_{\eta-k+1}$, маємо:

$$g_\eta + (k-1)\beta^{(1)} = g_\eta +$$

$$+(k-1)\beta^{(1)} - (k-|\omega|)\beta^{(1)} < (g_\eta + g_{\eta-1} + \ldots + g_{\eta-k+1}) -$$

$$-(g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + \ldots + g_{\eta-k+1}) = g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}.$$

У силу умови (3.14), кожна з нерівностей даних спілок, яка не містить x_ω, у точці $x^{(1)}$ перетворюється у строгу нерівність

$$|\omega|\beta^{(1)} < g_\eta + g_{\eta-1} + \ldots + g_{\eta-|\omega|+1}.$$

Таким чином, у характеристичній точці $x^{(1)}$ $(k-2)$ – площини, яка визначається нерівністю (3.34) спілки $m=1$, усі нерівності підсистеми (3.2), за виключенням нерівності (3.34), перетворюється в строгі.

Розглянемо нарешті випадок, коли $m = k$. Виберемо у цій спілці довільну нерівність

$$x_1 + x_2 + ... x_k \le g_\eta + g_{\eta-1} + ... + g_{\eta-k+1}. \quad (3.37)$$

Нехай $x^{(2)} = \left(x_1^{(2)}, x_2^{(2)}, ..., x_k^{(2)}\right)$ – характеристична точка $(k-2)$ – площини, яка визначається нерівністю (3.37). Позначимо $\left(g_\eta + g_{\eta-1} + ... + g_{\eta-k+1}\right)/k = \alpha^{(1)}$. Тоді $x_1^{(2)} = x_2^{(2)} = ... = x_k^{(2)} = \alpha^{(1)}$. Оскільки виконується умова $1 < \eta_n < k$, то:

$$g_{\eta-k+1} < \alpha^{(1)} < g_\eta. \quad (3.38)$$

Отже, існує таке t_4, для якого має місце співвідношеня:

$$g_1 \le ... \le g_{t_4} < \alpha^{(1)} \le g_{t_4+1} \le g_{t_4+2} \le ... \le g_\eta, \quad (3.39)$$

причому, $1 < t_4 < k$.

Покажемо, що в точці $x^{(2)}$ усі нерівності, за виключенням нерівності (3.37), будь-якої спілки $|\omega| \in J_k$ перетворюються в строгі. Для цього, як і раніше, розглянемо по черзі такі випадки: 1) $1 < |\omega| \le t_4$, 2) $t_4 < |\omega| < k$, 3) $|\omega| = k$.

1. Нехай $1 < (\omega(\le t_4$. При цій умові із співвідношення (3.39) безпосередньо випливає, що всі нерівності перетворюються в точці x в строгі числові нерівності $|\omega|\alpha^{(1)} < g_\eta + g_{\eta-1} + ... + g_{\eta-|\omega|+1}$.

2. Якщо $t_4 < |\omega| < k$, то нерівності в точці x набувають вигляду: $|\omega|\alpha^{(1)} \le g_\eta + g_{\eta-1} + ... + g_{\eta-|\omega|+1}$. За умови $t_4 < |\omega| < k$ із співвідношень (3.38), (3.39) випливає строга нерівність:

$$\left(k - |\omega|\right)\alpha^{(1)} > g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + ... + g_{\eta-k+1}.$$

Оскільки $k\alpha^{(1)} = g_\eta + g_{\eta-1} + ... + g_{\eta-k+1}$, то:

$$|\omega|\alpha^{(1)} = k\alpha^{(1)} - \left(k - |\omega|\right)\alpha^{(1)} < k\alpha^{(1)} - \left(g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + ... + g_{\eta-k+1}\right) =$$
$$= g_\eta + g_{\eta-1} + ... + g_{\eta-k+1} - \left(g_{\eta-|\omega|} + g_{\eta-|\omega|-1} + ... + g_{\eta-k+1}\right) =$$
$$= g_\eta + g_{\eta-1} + ... + g_{\eta-|\omega|+1}.$$

3. У спілці $|\omega| = k$ обмеження (3.37) у точці $x^{(2)}$ перетворюється у рівність $k\alpha^{(1)} = g_\eta + g_{\eta-1} + ... + g_{\eta-k+1}$. Ліва частина будь-якої іншої нерівності цієї спілки в точці $x^{(2)}$ набуває вигляду: $(k-1)\alpha^{(1)}$. Оскільки виконується умова (3.38), то:

$$(k-1)\alpha^{(1)} < k\alpha^{(1)} = g_\eta + g_{\eta-1} + ... + g_{\eta-k+1}.$$

Отже, в характеристичній точці $x^{( )}$ $(k-2)$ – площини, яка визначається нерівністю (3.37) спілки $m = k$, усі нерівності, за виключенням (3.37), будь-якої спілки $|\omega| \in J_k$, перетворюються в строгі.

Доведемо, що існує така точка, яка задовольняє усі обмеження підсистеми (3.2), крім вибраної нерівності, тобто виключення з підсистеми (3.2) вибраної нерівності приводить до зміни многогранника $\Pi^k_{\eta n}(G)$.

Виберемо довільну нерівність:

$$x_{\alpha_1} + x_{\alpha_2} + ... + x_{\alpha_m} \le g_\eta + g_{\eta-1} + ... + g_{\eta-m+1} \quad (3.40)$$

довільної спілки $m \in J_k$, відмінної від спілок (3.4), підсистеми (3.2). Нехай $x^* = \left(x_1^*, x_2^*, ..., x_k^*\right)$ – характеристична точка $(k-2)$ – площини, яка визначається нерівністю (3.40). У точці x^* усі нерівності спілок $|\omega| \in J_k$, за виключенням нерівності (3.40) перетворюються в строгі числові нерівності, що доведено вище. Нехай p-кількість таких нерівностей. (Оскільки підсистема (3.2) містить $2^k - 1$ обмежень, то $p = 2^k - 2$). Пронумеруємо ці строгі числові нерівності та позначимо $X_s^* < G_s^*$ ту з них, номер якої s. Позначимо

$$\left(g_\eta + g_{\eta-1} + ... + g_{\eta-m+1}\right) / m = \alpha^*,$$

$$\left(g_{\eta-m} + g_{\eta-m-1} + ... + g_{\eta-k+1}\right) / (k-m) = \beta^*.$$

Тоді $x_1^* = x_2^* = ... = x_k^* = \alpha^*$, $x_{m+1}^* = x_{m+2}^* = ... = x_k^* = \beta^*$. Зрозуміло, що $\forall s \in J_p \; \exists \varepsilon(s) > 0 : X_s^* - \varepsilon(s) < G_s^*$. Нехай $\varepsilon = \min_{s \in J_p}\{\varepsilon(s)\}$. Розглянемо підсистему (3.2) у точці $\bar{x} = \left(\bar{x}_1, \bar{x}_2, ..., \bar{x}_k\right)$, де

$$\bar{x}_j = x_j^* - \varepsilon / m, \forall j \in J_m, \; \bar{x}_j = x_j^* + \varepsilon / (k-m), \forall j \in J_k \setminus J_m,$$

тобто

$$\bar{x}_j = \alpha^* - \varepsilon / m, \forall j \in J_m, \bar{x}_j = \alpha_j^* + \varepsilon /(k-m), \forall j \in J_k \setminus J_m.$$

У цій точці нерівність (3.40) не виконується, оскільки:

$$\bar{x}_1 + \bar{x}_2 + ... + \bar{x}_m = m(\alpha^* - \varepsilon / m) = m\alpha^* - \varepsilon > m\alpha^* = g_\eta + g_{\eta-1} + ... + g_{\eta-m+1}$$

Усі інші обмеження підсистеми (3.2) у точці $\bar{x}$, в силу вибору ε, виконуються як строгі нерівності. Отже, точка $\bar{x}$ задовольняє всі обмеження підсистеми (3.2), крім нерівності (3.40), а тому виключення з підсистеми (3.2) цієї нерівності приводить до зміни многогранника $\Pi_{\eta n}^k(G)$, тобто, нерівність (3.40) не є надлишковим обмеженням.

З довільності вибору m-спілки, відмінної від спілок (3.4), та довільності вибору у цій спілці нерівності (3.40) випливає справедливість теореми для підсистеми (3.2). Теорему доведено.

Розглянемо приклад побудови незвідної системи обмежень загального многогранника розміщень.

Нехай $G = \{2,\ 2,\ 5,\ 5,\ 8,\ 8,\ 8\}$, $k = 5$. Маємо $S(G) = \{2,\ 5,\ 8\}$; $\eta = 7,\ \ \eta_1 = 2,\ \ \eta_n = 3$.

Тут $k > \eta - \eta_n$. Система лінійних обмежень, що описує загальний многогранник розміщень, запишеться наступним чином.

При $|\omega| = 1$ маємо нерівності:

$$x_1 \ge 2, \quad x_2 \ge 2, \quad x_3 \ge 2, \quad x_4 \ge 2, \quad x_5 \ge 2; \qquad (3.41)$$

$$x_1 \le 8, \quad x_2 \le 8, \quad x_3 \le 8, \quad x_4 \le 8, \quad x_5 \le 8. \qquad (3.42)$$

При $|\omega| = 2$ в системі такі нерівності:

$$\begin{aligned}
&x_1 + x_2 \ge 2+2,\ x_2 + x_4 \ge 2+2,\ x_1 + x_5 \ge 2+2,\ x_3 + x_5 \ge 2+2,\\
&x_1 + x_3 \ge 2+2,\ x_2 + x_5 \ge 2+2,\ x_2 + x_3 \ge 2+2,\ x_4 + x_5 \ge 2+2; \qquad (3.43)\\
&x_1 + x_4 \ge 2+2,\ x_3 + x_4 \ge 2+2,
\end{aligned}$$

$$\begin{aligned}
&x_1 + x_2 \le 8+8,\ x_2 + x_4 \le 8+8,\ x_1 + x_5 \le 8+8,\ x_3 + x_5 \le 8+8,\\
&x_1 + x_3 \le 8+8,\ x_2 + x_5 \le 8+8,\ x_2 + x_3 \le 8+8,\ x_4 + x_5 \le 8+8. \qquad (3.44)\\
&x_1 + x_4 \le 8+8,\ x_3 + x_4 \le 8+8,
\end{aligned}$$

При $|\omega| = 3$ маємо:

$$\begin{aligned}
&x_1 + x_2 + x_3 \geq 2+2+5, &\quad x_1 + x_4 + x_5 \geq 2+2+5,\\
&x_1 + x_2 + x_4 \geq 2+2+5, &\quad x_2 + x_3 + x_4 \geq 2+2+5,\\
&x_1 + x_2 + x_5 \geq 2+2+5, &\quad x_2 + x_3 + x_5 \geq 2+2+5,\\
&x_1 + x_3 + x_4 \geq 2+2+5, &\quad x_2 + x_4 + x_5 \geq 2+2+5,\\
&x_1 + x_3 + x_5 \geq 2+2+5, &\quad x_3 + x_4 + x_5 \geq 2+2+5;
\end{aligned} \tag{3.45}$$

$$\begin{aligned}
&x_1 + x_2 + x_3 \leq 8+8+8, &\quad x_1 + x_4 + x_5 \leq 8+8+8,\\
&x_1 + x_2 + x_4 \leq 8+8+8, &\quad x_2 + x_3 + x_4 \leq 8+8+8,\\
&x_1 + x_2 + x_5 \leq 8+8+8, &\quad x_2 + x_3 + x_5 \leq 8+8+8,\\
&x_1 + x_3 + x_4 \leq 8+8+8, &\quad x_2 + x_4 + x_5 \leq 8+8+8,\\
&x_1 + x_3 + x_5 \leq 8+8+8, &\quad x_3 + x_4 + x_5 \leq 8+8+8.
\end{aligned} \tag{3.46}$$

Якщо $|\omega| = 4$, то:

$$\begin{aligned}
&x_1 + x_2 + x_3 + x_4 \geq 2+2+5+5,\\
&x_1 + x_2 + x_3 + x_5 \geq 2+2+5+5,\\
&x_1 + x_2 + x_4 + x_5 \geq 2+2+5+5,\\
&x_1 + x_3 + x_4 + x_5 \geq 2+2+5+5,\\
&x_2 + x_3 + x_4 + x_5 \geq 2+2+5+5;
\end{aligned} \tag{3.47}$$

$$\begin{aligned}
&x_1 + x_2 + x_3 + x_4 \leq 8+8+8+5,\\
&x_1 + x_2 + x_3 + x_5 \leq 8+8+8+5,\\
&x_1 + x_2 + x_4 + x_5 \leq 8+8+8+5,\\
&x_1 + x_3 + x_4 + x_5 \leq 8+8+8+5,\\
&x_2 + x_3 + x_4 + x_5 \leq 8+8+8+5.
\end{aligned} \tag{3.48}$$

Для $|\omega| = 5$ отримуємо:

$$x_1 + x_2 + x_3 + x_4 + x_5 \geq 2+2+5+5+8;$$

$$x_1 + x_2 + x_3 + x_4 + x_5 \leq 8+8+8+5+5. \tag{3.49}$$

За теоремою надлишковими серед нерівностей підсистеми (3.1) є нерівності сполук $2 \leq i \leq \eta_1$; $\eta - \eta_n \leq i \leq k-1$, тобто сполук 2 і 4 – системи (3.43), (3.47), а надлишковими серед нерівностей підсистеми (3.2) є нерівності спілок $2 \le i \le \eta_n$, тобто спілок 2, 3 – системи (3.44), (3.46). Таким чином незвідна система складається з нерівностей систем (3.41), (3.42), (3.45), (3.48), (3.49).

3.2. Незвідна система обмежень для многогранника полірозміщень

Якщо зафіксувати $i \in J_s$, то з системи (1.17) одержимо систему:

$$\sum_{j=1}^{|\omega^i|} g_j^{N_i} \le \sum_{j \in \omega^i} x_j \le \sum_{j=1}^{|\omega^i|} g_{\eta_i - j+1}^{N_i} \quad \forall \omega^i \subset N_i', \qquad (3.50)$$

яка описує загальний многогранник розміщень $\Pi_{\eta_i n_i}^{k_i}(G^{N_i})$, де n_i – кількість елементів основи мультимножини G^{N_i}.

Лема 3.3[76]. Многогранник (1.17) є добутком многогранників (3.50) $\forall i \in J_s$, тобто $\Pi_{\eta n}^{k}(G,\ H) = \bigotimes_{i=1}^{s} \Pi_{\eta_i n_i}^{k_i}(G^{N_i})$.

Теорема 3.4. Якщо існує $i \in J_s$, для якого $\eta_1^i > 1$ або $\eta_{n_i}^i > 1$, то загальний многогранник полірозміщень $\Pi_{\eta n}^{ks}(G,H)$:

$$\begin{cases} \sum_{j=1}^{|\omega^i|} g_j^{N_i} \le \sum_{j \in \omega^i} x_j, & (3.51) \\ \sum_{j \in \omega^i} x_j \le \sum_{j=1}^{|\omega^i|} g_{n_i - j+1}^{N_i}, & (3.52) \end{cases}$$

має надлишкові обмеження:

при $\eta_1^i > 1$ – усі нерівності спілок:

$$(i,\ 2),\ (i,\ 3),\ \ldots,\ (i, \eta_1^i),\ (i,\ \eta^i - \eta_{n_i}^i),\ \ldots,\ (i,\ k_i - 1) \quad (3.53)$$

серед нерівностей (3.51),
а при $\eta_{n_i}^i > 1$ – усі нерівності спілок:

$$(i,\ 2),\ (i,\ 3),\ \ldots,\ (i, \eta_{n_i}^i),\ (i,\ \eta^i - \eta_1^i),\ \ldots,\ (i,\ k_i - 1) \quad (3.54)$$

серед нерівностей (3.52).

Виключення з системи (3.51), (3.52) нерівностей (3.53) і (3.54) перетворює її на незвідну систему обмежень загального многогранника полірозміщень.

Доведення. За теоремою про незвідність системи обмежень загального многогранника розміщень маємо, що якщо в первинній специфікації мультимножини G^{N_i} $\eta_1^i > 1$, то надлишковими обмеженнями многогранника (3.50) (або, що те ж саме, (3.51), (3.52) при фіксованому значенні $i \in J_s$) є усі нерівності системи (3.51) спілок $(i,\ t)$, де $2 \le t \le \eta_1^i$ або $\eta^i - \eta_{n_i}^i \le t \le k_i - 1$, а при $\eta_{n_i}^i > 1$ – усі нерівності системи (3.52) спілок $(i,\ t)$, де $2 \le t \le \eta_{n_i}^i$ або $\eta^i - \eta_1^i \le t \le k_i - 1$ і тільки вони. Виключення вказаних надлишкових нерівностей дає незвідну систему обмежень многогранника (3.50).

Оскільки система (2.12) (або (3.51), (3.52)) повністю вичерпується системами вигляду (3.50) за умови $i = 1,\ 2,\ \dots,\ s$, то твердження доведено.

Розглянемо приклад побудови незвідної системи обмежень загального многогранника полірозміщень.

Нехай $G = \{1,\ 1,\ 3,\ 3,\ 4,\ 4,\ 4,\ 4,\ 7,\ 7,\ 7,\ 8,\ 8,\ 8,\ 8\}$, $\eta = 15$. Маємо $S(G) = \{1,\ 3,\ 4,\ 7,\ 8\}$, $n = 5$, $[G] = \{2,\ 2,\ 4,\ 3,\ 4\}$.

Нехай $k = 10$, $s = 3$. Розіб'ємо $J_\eta = J_{15} = \{1,\ 2,\ \dots,\ 15\}$ на s множин:

1) $N_1 = \{1,\ 2,\ 3,\ 5\}$, $|N_1| = 4$, $G^{N_1} = \{1,\ 1,\ 3,\ 4\}$, $\eta_1^1 = 2$, $\eta_{n_1}^1 = \eta_4^1 = 1$.

2) $N_2 = \{4,\ 12,\ 13,\ 14,\ 15\}$, $|N_2| = 5$, $G^{N_2} = \{3,\ 8,\ 8,\ 8,\ 8\}$, $\eta_1^2 = 1$, $\eta_{n_2}^2 = \eta_5^2 = 4$.

3) $N_3 = \{6,\ 7,\ 8,\ 9,\ 10,\ 11\}$, $|N_3| = 6$, $G^{N_3} = \{4,4,4,7,7,7\}$, $\eta_1^3 = 3$, $\eta_{n_3}^3 = \eta_6^3 = 3$.

Розіб'ємо число k на s доданків. Нехай $k = 2 + 4 + 4$. Тоді

1) $k_1 = 2$; $\eta_1^1 = 2$; $\eta_{n_1}^1 = 1$; $N_1' = \{0+1, 0+2, \dots, k_1\} = \{1,2\}$,

$$\omega^1 = \{1\}, \{2\}, \{1,2\}.$$

2) $k_2 = 4$; $\eta_1^2 = 1$; $\eta_{n_2}^2 = 4$; $N_2' = \{k_1+1, k_1+2, \dots, k_1+k_2\} = \{3,4,5,6\}$,

$$\omega^2 = \{3\}, \{4\}, \{5\}, \{6\}, \{3,4\}, \{3,5\}, \{3,6\}, \{4,5\}, \{4,6\}, \{5,6\}, \{3,4,5\}, \{3,4,6\},$$
$$\{3,5,6\}, \{4,5,6\}, \{3,4,5,6\}.$$

3) $k_3 = 4$; $\eta_1^3 = 3$; $\eta_{n_3}^3 = 3$; $N_3' = \{k_1 + k_2 + 1, k_1 + k_2 + 2, ..., k_1 + k_2 + k_3\} =$
$= \{7,8,9,10\}$,

$$\omega^3 = \{7\},\{8\},\{9\},\{10\},\{7,8\},\{7,9\},\{7,10\},\{8,9\},\{8,10\},\{9,10\},\{7,8,9\},$$
$$\{7,8,10\},\{7,9,10\},\{8,9,10\},\{7,8,9,10\}.$$

Формуємо систему обмежень

Спілка (1,1) (i=1, $|\omega| = 1$):

$$x_1 \geq 1;\ \ x_2 \geq 1;\ \ x_1 \leq 4;\ \ x_2 \leq 4.$$

Спілка (1,2) (i=1, $|\omega| = 2$):

$$x_1 + x_2 \geq 1 + 1;$$
$$x_1 + x_2 \leq 4 + 3. \qquad (3.55)$$

Спілка (2,1) (i=2, $|\omega| = 1$):

$$x_3 \geq 3;\ \ x_4 \geq 3;\ \ x_5 \geq 3;\ \ x_6 \geq 3; \qquad (3.56)$$
$$x_3 \leq 8;\ \ x_4 \leq 8;\ \ x_5 \leq 8;\ \ x_6 \leq 8.$$

Спілка (2,2) (i=2, $|\omega| = 2$):

$$\begin{aligned} &x_3 + x_4 \geq 3 + 8; x_3 + x_5 \geq 3 + 8; x_3 + x_6 \geq 3 + 8;\\ &x_4 + x_5 \geq 3 + 8; x_4 + x_6 \geq 3 + 8; x_5 + x_6 \geq 3 + 8;\\ &x_3 + x_4 \leq 8 + 8; x_3 + x_5 \leq 8 + 8; x_3 + x_6 \leq 8 + 8;\\ &x_4 + x_5 \leq 8 + 8; x_4 + x_6 \leq 8 + 8; x_5 + x_6 \leq 8 + 8. \end{aligned} \qquad (3.57)$$

Спілка (2,3) (i=2, $|\omega| = 3$):

$$\begin{aligned} &x_3 + x_4 + x_5 \geq 3 + 8 + 8; x_3 + x_4 + x_6 \geq 3 + 8 + 8;\\ &x_3 + x_5 + x_6 \geq 3 + 8 + 8; x_4 + x_5 + x_6 \geq 3 + 8 + 8;\\ &x_3 + x_4 + x_5 \leq 8 + 8 + 8; x_3 + x_4 + x_6 \leq 8 + 8 + 8;\\ &x_3 + x_5 + x_6 \leq 8 + 8 + 8; x_4 + x_5 + x_6 \leq 8 + 8 + 8. \end{aligned} \qquad (3.58)$$

Спілка (2,4) (i=2, $|\omega| = 4$):

$$x_3 + x_4 + x_5 + x_6 \geq 3 + 8 + 8 + 8;$$
$$x_3 + x_4 + x_5 + x_6 \leq 8 + 8 + 8 + 8. \qquad (3.59)$$

Спілка (3,1) (i=3, $|\omega| = 1$):

$$\begin{aligned} &x_7 \geq 4; \quad x_8 \geq 4; \quad x_9 \geq 4; \quad x_{10} \geq 4; \\ &x_7 \leq 7; \quad x_8 \leq 7; \quad x_9 \leq 7; \quad x_{10} \leq 7. \end{aligned} \tag{3.60}$$

Спілка (3,2) (i=3, $|\omega| = 2$):

$$\begin{aligned} &x_7 + x_8 \geq 4 + 4; x_7 + x_9 \geq 4 + 4; x_7 + x_{10} \geq 4 + 4; \\ &x_8 + x_9 \geq 4 + 4; x_8 + x_{10} \geq 4 + 4; x_9 + x_{10} \geq 4 + 4; \\ &x_7 + x_8 \leq 7 + 7; x_7 + x_9 \leq 7 + 7; x_7 + x_{10} \leq 7 + 7; \\ &x_8 + x_9 \leq 7 + 7; x_8 + x_{10} \leq 7 + 7; x_9 + x_{10} \leq 7 + 7. \end{aligned} \tag{3.61}$$

Спілка (3,3) (i=3, $|\omega| = 3$):

$$\begin{aligned} &x_7 + x_8 + x_9 \geq 4 + 4 + 4; \quad x_7 + x_8 + x_{10} \geq 4 + 4 + 4; \\ &x_8 + x_9 + x_{10} \geq 4 + 4 + 4; \quad x_7 + x_9 + x_{10} \geq 4 + 4 + 4; \\ &x_7 + x_8 + x_9 \leq 7 + 7 + 7; \quad x_7 + x_8 + x_{10} \leq 7 + 7 + 7; \\ &x_8 + x_9 + x_{10} \leq 7 + 7 + 7; \quad x_7 + x_9 + x_{10} \leq 7 + 7 + 7. \end{aligned} \tag{3.62}$$

Спілка (3,4) (i=3, $|\omega| = 4$):

$$x_7 + x_8 + x_9 + x_{10} \geq 4 + 4 + 4 + 7; \; x_7 + x_8 + x_9 + x_{10} \leq 7 + 7 + 7 + 4 .$$

Виключення надлишкових обмежень (3.55) – (3.62) перетворює наведену систему обмежень в незвідну систему. Многогранник полірозміщень $\Pi_{15,\,5}^{10,\,3}(G)$ є добутком многогранників розміщень:

$$\Pi_{15,\,5}^{10,\,3}(G) = \bigotimes_{i=1}^{3} \Pi_{\eta_i n_i}^{k_i}(G^{N_i}) = \Pi_{4,\,3}^{2}(G^{N_1}) \otimes \Pi_{5,\,2}^{4}(G^{N_2}) \otimes \Pi_{6,\,2}^{4}(G^{N_3}),$$

кожен з яких задається незвідною системою вигляду (3.50).

РОЗДІЛ 4
ЗАДАЧА РОЗМІЩЕНЬ ОБ'ЄКТІВ ОБСЛУГОВУВАННЯ ЯК ЗАДАЧА ЕВКЛІДОВОЇ ПОЛІКОМБІНАТОРНОЇ ОПТИМІЗАЦІЇ ТА МЕТОДИ ЇЇ РОЗВ'ЯЗУВАННЯ

4.1. Постановка задачі та її математична модель

Велика кількість важливих задач проектування і розміщення добре описуються за допомогою моделей оптимізаційних задач на комбінаторних множинах. При побудові математичної моделі розглянутої в даному розділі задачі застосовано апарат евклідової комбінаторної оптимізації [24; 58; 66], зокрема властивості евклідових комбінаторних множин переставлень та поліроміщень.

Розглянемо задачу розміщення об'єктів обслуговування.

Нехай задана квадратна матриця відстаней між r пунктами, елемент якої r_{ij} -відстані між пунктами i і j.

Відома кількість клієнтів a_i у кожному населеному пункті і можливі ємності об'єктів обслуговування $q_1, \; q_2, \; ..., \; q_\omega$, які можна використовувати для обслуговування. Тут ω-кількість різних ємностей. Необхідно так розмістити об'єкти обслуговування в пунктах, щоб вони були повністю завантажені обслуговуванням клієнтів, які знаходяться в заданому радіусі обслуговування ρ_{qi} з мінімізацією кількості клієнтів, які не обслуговуються (у випадку ієрархічної системи їх обслуговування передається на вищий рівень, на якому ставиться аналогічна задача). Розглядаємо задачу на фіксованому рівні. Вважаємо, не обмежуючи загальності міркувань, що $a_1 \ge a_2 \ge ... \ge a_r$. Будемо вважати, що $\rho_{qi} = \rho = const$.

Дана задача виникла в архітектурному проектуванні при розміщенні лікувальних закладів в одному з районів Полтавської області [175]. Для її розв'язування архітекторами використовувався евристичний алгоритм, який давав наближений розв'язок.

Побудуємо математичну модель цієї задачі у вигляді задачі евклідової комбінаторної оптимізації з використанням властивостей множин переставлень і поліроміщень.

Введемо вектор x, складений з s векторів x^i (кожний довжини r), елементами x_i^j яких є кількість клієнтів a пункту j, які обслуговуються в об'єкті Γ_i, тобто

$$x_i^j = (x_1^1, ..., x_r^1, x_1^2, ..., x_r^2, ..., x_1^\tau, ..., x_r^\tau, ..., x_{(s-1)r+1}^s, ..., x_{rs}^s,), \; i \in J_s .$$

Зауважимо, що $x_i^j = a_i$, якщо пункт j обслуговується об'єктом Γ_i. В іншому випадку $x_i^j = 0$. Очевидно, що $\sum_{j=1}^{s}\sum_{i=1}^{r} x_i^j = \sum_{i=1}^{r} a_i$.

Вектор x має довжину $k = s \cdot r$ (s – кількість об'єктів обслуговування, які планується розмістити, r – кількість пунктів). Якщо це число не задано, а підраховується при розв'язуванні задачі, то можна покласти $s = r$.

Нехай $G = \{0, ..., 0, a_r, ..., a_1\} = \{g_1, ..., g_k\}$ – множина, що має $\nu \le r+1$ різних елементів $g_1 \le ... \le g_k$, $g_1 = g_2 = ... = g_{k-r} = 0$, $g_1 \le ... \le g_k$, $g_{k-r+1} = a_r, ..., g_k = a_1$. Очевидно, якщо розглянути множину переставлень $E_{k\nu}(G)$ з елементів мультимножини G, то ми розглянемо всі можливі включення кожного пункту в зону обслуговування кожного об'єкту як вибір $x \in E_{k\nu}(G)$.

Введемо до розгляду вектор:

$$y^j = (y_1^j, ..., y_\omega^j),$$

$$j \in J_s, \; y_i^j \in \{0 \cdot q_i, 1 \cdot q_i, 2 \cdot q_i, ..., k(i) \cdot q_i\} = Q_i,$$

де $k(i)$ – кількість однакових $q(i)$ об'єктів у пункті j.

Якщо ціле число $k(i)$ невідоме, то його можна обчислити за формулою:

$$k(i) \cdot q_i \le \sum_{j=1}^{r} a_j, \qquad (k(i)+1) \cdot q_i > \sum_{j=1}^{r} a_j,$$

де a_j – кількість клієнтів, які попали в радіус обслуговування ρ.

Вектор $y = (y^1, ..., y^s)$ можна розглядати як елемент множини полірозміщень, в якій кожне полірозміщення має ω елементів, де $\omega = k_1 + k_2 + ... + k_\omega$, $k_i = 1$ $\forall i \in J_Q$, $Q = Q_1 + ... + Q_\omega$. Цю множину полірозміщень позначимо $E_{\eta n}^{\omega\omega}(Q, H)$, де множина $H = \{\pi = (\pi_1, ..., \pi_\omega) \mid \pi_i \in Q_i\}$, $\eta = \sum_{i=1}^{\omega} k(i) + 1$, n – кількість різних елементів у Q.

Нехай E – полікомбінаторна множина з елементів $z = (x, y)$, де $x \in E_{k\nu}(G)$, $y \in E_{\eta n}^{\omega\omega}(Q, H)$, тобто $E = E_{k\nu}(G) \times E_{\eta n}^{\omega\omega}(Q, H)$.

Тоді математична модель задачі, що розглядається, буде мати вигляд:

Знайти

$$\min_{z\in E}\sum_{j=1}^{s}\left(\sum_{i\in J_r, r_{ij}\le\rho}^{r} x_i^j-\sum_{i=1}^{\omega} y_i^j\right) \tag{4.1}$$

при обмеженнях:

$$\sum_{i\in J_r, r_{ij}\le\rho}^{r} x_i^j-\sum_{i=1}^{\omega} y_i^j\ge 0,\quad \forall j\in J_s. \tag{4.2}$$

Задача (4.1), (4.2) є задачею евклідової комбінаторної оптимізації. В наступних підрозділах розглянуті методи її розв'язування, які ґрунтуються на використанні властивостей евклідових комбінаторних та полікомбінаторних множин.

4.2. Застосування методу гілок і меж до розв'язування задачі розміщення об'єктів обслуговування

Даний підхід ґрунтується на тому, що оптимальна забудова відділень відбувається у випадку, коли на кожному кроці поетапної забудови об'єктів мінімізується кількість не обслугованих клієнтів. Це дає можливість визначити умову, яка може слугувати оцінкою для множини допустимих розв'язків в методі гілок і меж, в якому кількість рівнів галуження дорівнює кількості населених пунктів, в яких планується розмістити мережу об'єктів обслуговування, тобто на кожному рівні відбувається забудова об'єктів в одному з населених пунктів. Вказана оцінка має вигляд:

$$v^j(t)=v^{j-1}(t)+\min_{i\in J_{m_j}}\left(\sum_{i=1}^{r}x_i^j-\sum_{i=1}^{\omega}y_i^j\right), \tag{4.3}$$

де j – рівень галуження, $j\in J_s$,

m_j – кількість елементів, які розглядаються на j-му рівні галуження.

При переході від одного рівня до іншого проводимо галуження елемента з мінімальною оцінкою (4.3).

Алгоритм:

Крок 1. Формуємо множину векторів x^j, що задовольняє умовам:

1) якщо $r_i^j > \rho$, то $x_i^j = 0$;

2) якщо $r_i^j \le \rho$, то x_i^j вибираємо з множини $\{0, a\}$.

Крок 2. Для кожного вектора x^j визначаємо вектор y^j, що задовольняє умовам:

1) $y_i^j \in Q_i, \forall i \in J_\omega$;

2) різниці $(\sum_{i=1}^{r} x_i^j - \sum_{i=1}^{\omega} y_i^j)$ є мінімальними для всіх векторів x^j.

Ці різниці служать оцінками (4.3) на першому рівні галуження.

Крок 3. Розгалужуємо вектор x^j, який має мінімальну оцінку серед усіх векторів даного рівня галуження. Результатом галуження є всі вектори x^{j+1}, для яких з умови $x_i^j = a_i$ випливає $x_i^{j+1} = 0$, $i \in J_r$, $j \in J_s$.

Крок 4. Обчислюємо оцінки за формулою (4.3).

Крок 5. Якщо $j < s$, то переходимо до пункту 3. В іншому випадку визначаємо вектор з мінімальною оцінкою на рівні s.

Крок 6. Оптимальний розв'язок задачі визначається множиною векторів x^j, яким відповідають мінімальні оцінки на кожному рівні галуження.

Крок 7. Обчислюємо значення цільової функції (табл. 4.1).

Розглянемо приклад застосування алгоритму методу гілок і меж до даної задачі.

Нехай задана кількість населених пунктів $r = 4$. Відомо, що кількість клієнтів, які потребують обслуговування для кожного населеного пункту відповідно дорівнює $a_1 = 14$, $a_2 = 8$, $a_3 = 7$ і $a_4 = 6$ клієнтів. Задано радіус обслуговування $\rho = 25$ та матрицю відстаней між населеними пунктами i та j:

$$r_{ij} = \begin{pmatrix} 0 & 20 & 30 & 28 \\ & 0 & 12 & 35 \\ & & 0 & 40 \\ & & & 0 \end{pmatrix}.$$

Дано, що в населених пунктах необхідно розмістити об'єкти двох типів $q = 14$, $q_2 = 6$ таким чином, щоб мінімізувати кількість клієнтів, які не зможуть обслуговуватися на даному рівні.

Кроки 1,2 поданы в таблицы 4.1.

Таблиця 4.1. Перший рівень галуження за методом гілок і меж

№ п/п	x^1	y^1	$v^1(t) = \min_{t \in J_{m_1}}(\sum_{i=1}^{r} x_i^j - \sum_{i=1}^{\omega} y_i^j)$
1	(14, 0, 0, 0)	(0, 14)	0
2	(14, 8, 0, 0)	(6, 14)	2
3	(0, 8, 0, 0)	(6, 0)	2
4	(14, 8, 7, 0)	(0, 28)	1
5	(14, 0, 7, 0)	(6, 14)	1
6	(0, 0, 7, 0)	(6, 0)	1
7	(0, 8, 7, 0)	(0, 14)	1

Кроки 3, 4, 5. Мінімальну оцінку має вектор $x^1 = (14, 0, 0, 0)$. Переходимо до другого рівня галуження ($j = 2$) (табл. 4.2):

Таблиця 4.2. Другий рівень галуження за методом гілок і меж

№ п/п	x^2	$v^2(t)$
1.	(0, 8, 0, 0)	2
2.	(0, 0, 7, 0)	1
3.	(0, 8, 7, 0)	1

На другому рівні маємо два вектори з мінімальними оцінками. Вибираємо, наприклад, вектор $x^2 = (0, 8, 7, 0)$. Переходимо до третього рівня галуження ($j = 3$).

На третьому рівні галуження маємо порожню множину допустимих векторів, оскільки клієнти із 1, 2 і 3 пунктів, які б могли бути обслуговані у третьому пункті, вже розміщені на обслуговування на попередніх рівнях. Клієнти з четвертого населеного пункту не можуть обслуговуватися у третьому пункті в силу недосяжності радіусу обслуговування ρ. Відсутності обслуговування відповідає нульовий вектор (0, 0, 0, 0).

Розгалужуємо вектор $x^3 = (0, 0, 0, 0)$. Переходимо до останнього рівня галуження ($j = 4$): $x^4 = (0, 0, 0, 6)$.

Крок 6. Таким чином, оптимальне розміщення об'єктів обслуговування визначає множина векторів {(14, 0, 0, 0), (0, 8, 7, 0), (0, 0, 0, 0), (0, 0, 0, 6)}. На базі цієї множини можна утворити два оптимальних розв'язки:

$$X_1^* = (14,\ 0,\ 0,\ 0;\ \ 0,\ 8,\ 7,\ 0;\ \ 0,\ 0,\ 0,\ 0;\ \ 0,\ 0,\ 0,\ 6),$$

$$X_2^* = (14,\ 0,\ 0,\ 0;\ \ 0,\ 0,\ 0,\ 0;\ \ 0,\ 8,\ 7,\ 0;\ \ 0,\ 0,\ 0,\ 6).$$

Клієнти будуть обслуговуватися в першому, четвертому, а також у другому (для X_1^*) або третьому (для X_2^*) пунктах; у першому пункті розташовано 1 об'єкт на 14 місць, у другому (для X_1^*) або третьому (для X_2^*) – 1 об'єкт на 14 місць, у четвертому – 1 об'єкт на 6 місць.

Крок 7. Значення цільової функції $f(X_1^*) = (14 - 14) + (15 - 14) + (0 - 0) + (6 - 6) = 1$; $f(X_2^*) = (14 - 14) + (0 - 0) + (15 - 14) + (6 - 6) = 1$. Отже, кількість клієнтів, які будуть обслуговуватися на вищому рівні при даному розміщенні об'єктів обслуговування дорівнює 1.

4.3. Застосування методів дискретної оптимізації для розв'язування задачі розміщення об'єктів обслуговування

4.3.1. Зв'язок між задачами в дискретній та комбінаторній постановках. Розглянута в п. 4.1 задача є задачею комбінаторної оптимізації, які належать до широкого класу задач дискретного програмування.

У [115, 116] показаний зв'язок між задачами евклідової комбінаторної оптимізації на полірозміщеннях і задачами цілочислового програмування. Це дало можливість розглядати задачі цілочислової оптимізації як задачі на загальній множині полірозміщень.

Доведено [116], що задача знаходження

$$\min_{x \in R^k} f_0(x) \tag{4.4}$$

при обмеженнях

$$x \in Z_k,\ a_j \le x_i \le b_j,\ a,b \in Z\ \ \forall i \in J_k, \tag{4.5}$$

$$f_j(x) \le d_j,\ d_j \in R^1,\ \forall j \in J_m, \tag{4.6}$$

де Z – множина цілих чисел,

Z_k – множина точок з цілочисловими координатами у просторі R^k,

$f_j(x)$ – функції, що діють з Z_k в R^1 $\forall j \in J_m^0$, $m \in Z$, $m \ge 0$, $x = (x_1, \ldots, x_k)$, еквівалентна наступній задачі: знайти (4.4) при обмеженнях

$$x \in E_{\eta n}^{ks}(G,H), \tag{4.7}$$

де $E_{\eta n}^{ks}(G,H)$ – загальна множина полірозміщень і додаткових обмеженнях (4.6).

Тут мультимножина G являє собою як суму мультимножин G_i:

$$G=\sum_{i=1}^{s} G_i=\{a_1, ..., b_1, a_2, ..., b_2, ..., a_m, ..., b_m\},$$

$G_i=S(G_i)=\{a_i, a_i+1, ..., b_i\}$, $|G_i|=n_i=b_i-a_i+1$, $[G_i]=\{1^{n_i}\}$ $\forall i\in J_s$. Приймається [116], що $k=s$, тобто $k_1=...=k_s=1$, а множина H утворюється з множин

$$N_1=\{1, ..., n_1\}, N_2=\{n_1+1, ..., n_1+n_2\},...,$$
$$N_s=\{n_1+...+n_{s-1}+1, ..., n_1+...+n_s\}.$$

Покажемо, що задача дискретної оптимізації [43, 132] може бути представлена як евклідова комбінаторна задача на множині полірозміщень.

Задача дискретного програмування [43, 132] має вигляд:

Знайти

$$\max_{x\in R^k} f(x) \tag{4.8}$$

при обмеженнях (4.6), якщо координати вектора $x=(x_1, x_2, ..., x_k)$, $x_i\geq 0$ обираються із відповідних множин дискретних значень:

$$x_j\in A^j=\left\{A_{j_1}, A_{j_2}, ..., A_{j_{m_j}}\right\}, \qquad j\in J_n. \tag{4.9}$$

Розглянемо множини $G_i=\left\{A_1^i, A_2^i, ..., A_{m_i}^i\right\}$, $|G_i|=m_i$, $[G_i]=\left\{1^{m_i}\right\}$ $\forall i\in J_k$, мультимножину $G=\sum_{i=1}^{s} G_i$ і множину H, що утворена як і в [116].

Тоді, задача (4.6), (4.8), (4.9) буде еквівалентною до задачі (4.6) – (4.8), де мультимножина G у (4.7) утворена описаним вище чином.

Математична модель задачі розміщення об'єктів обслуговування (4.1), (4.2) у вигляді задачі (4.6) – (4.8) буде така:

Максимізувати

$$F(x,y)=\sum_{j=1}^{s}\left(\sum_{i=1}^{\omega} y_i^j-\sum_{i=1}^{r} x_i^j\right) \tag{4.10}$$

при виконанні (4.2), а також при умовах

$$x\in E_{kv}(G),\ G=\sum_{i=1}^{s} G_i,\ G_i=\{0;\ a_i\}\ \forall i\in J_r \tag{4.11}$$

$$y\in E_{\eta n}^{\omega\omega}(Q,H),\ Q=\sum_{i=1}^{\omega} Q_i,\ Q_i=\{q_i, 2q_i, ..., k(i)q_i\}\ \forall i\in J_\omega \tag{4.12}$$

4.3.2. Алгоритм застосування методу Дальтона – Ллевеліна до розв'язування задачі. До задачі дискретного програмування у вигляді (4.6), (4.8), (4.9) може бути застосований метод Дальтона-Ллевеліна [43; 132], який полягає в наступному.

Нехай $x(G_k, F)$ – оптимальний розв'язок задачі (4.6), (4.8), (4.9), $\|x_{ij}^k\|$ – елементи відповідної симплекс-таблиці.

Якщо $x(G_k, F)$ є недопустимим розв'язком задачі (4.6), (4.8), (4.9) і $A_{iv} < x_{i0}^k < A_{iv+1}$ тоді, використовуючи i-й рядок симплекс-таблиці, можна побудувати правильне відсікання за формулами:

$$z_i = -\gamma_0^k + \sum_{j \in N_k} \gamma_j^k x_j, \tag{4.13}$$

$$z_i \geq 0, \tag{4.14}$$

де $\gamma_0^k = x_{i0}^k - A_{iv}$,

$$\gamma_j^k = \begin{cases} x_{ij}^k, x_{ij}^k \geq 0, \\ \dfrac{x_{i0}^k - A_{iv}}{A_{iv+1} - x_{i0}^k}\left(-x_{ij}^k\right), x_{ij}^k < 0. \end{cases} \tag{4.15}$$

Алгоритм Дальтона – Ллевеліна описується наступним чином [125].

1. Розв'язуємо (G_k, F) – задачу (4.6), (4.8) (на початку $k = 0$).

Нехай $x(G_k, F)$, $k = 0, 1, 2, \ldots$ оптимальний розв'язок (G_k, F) – задачі, $\|x^k\|$ — симплекс-таблиця.

2. Перевіряємо умову допустимості по всіх координатах оптимального вектора розв'язку $x(G_k, F)$ (G_k, F)-задачі. Якщо умова допустимості виконується, то одержаний розв'язок є оптимальним розв'язком вихідної задачі (4.6), (4.8), (4.9). Якщо умова допустимості не виконується хоча б по одній координаті, переходимо до п. 3.

3. Нехай x_{i0}^k, $i \in J_{n_1}$ не задовольняє умові допустимості. Тоді вибираємо

$$i_0 = min\{i \mid i \in J_{n_1}, \ \ x_{i0}^k - \text{не задовольняє (4.9)}\}.$$

4. Для вибраного номера $i = i_0$ будуємо правильне відсікання, тобто вводимо додаткову змінну

$$x_{n+k+1} \geq 0, \quad x_{n+k+1} = -\gamma_0^k + \sum_{j \in N_k} \left(-\gamma_j^k\right)\left(-x_j\right),$$

де γ_j^k визначається формулою (4.15).

5. Додаємо лінійне обмеження, що визначає правильне відсікання, до умов (G_k, F)-задачі і одержуємо нову (G_{k+1}, F)-задачу. Покладаємо $k=k+1$ і переходимо до п. 1.

Очевидно, що даний алгоритм може бути застосований до задачі у постановці (4.6) – (4.8) за умови лінійності цільової функції (4.8) і в цьому випадку може розглядатися як алгоритм, що розв'язує певну задачу евклідової комбінаторної оптимізації, а саме задачу (4.6) – (4.8).

Розглянемо приклад складання математичної моделі задачі розміщення об'єктів обслуговування у вигляді (4.2), (4.10) – (4.12), до розв'язування якої можливе застосування такого алгоритму.

Нехай маємо 3 населених пункти. Кількість клієнтів, що потребують обслуговування для кожного населеного пункту відповідно дорівнюють 10, 8 і 7 клієнтів. У даних населених пунктах необхідно розмістити об'єкти двох типів, які можуть обслуговувати відповідно 6 і 13 клієнтів, таким чином, щоб мінімізувати кількість клієнтів, які не зможуть обслуговуватися на даному рівні.

Для спрощення прикладу будемо вважати, що радіус обслуговування ρ достатньо великий, тобто кожен населений пункт попадає в зону обслуговування кожного з об'єктів.

Маємо, $a_1 = 10, a_2 = 8, a_3 = 7, q_1 = 6, q_2 = 13$.

Максимізувати:

$$F(x,y) = \sum_{j=1}^{3}\left(\sum_{i=1}^{2} y_i^j - \sum_{i=1}^{3} x_i^j\right)$$

при умовах

$$\sum_{i=1}^{3} x_i^j - \sum_{i=1}^{2} y_i^j \ge 0, \forall j \in J_3$$

$$x_i^j \ge 0, y_i^j \ge 0 \, \forall i \in J_3$$

$$x \in E_{6,4}(G),$$

де $G = G_1 + G_2 + G_3$; $G_1 = \{0,\ 10\}$, $G_2 = \{0,\ 8\}$, $G_3 = \{0,\ 7\}$.

$$y \in E_{8,5}^{2,2}(Q, H),$$

де $Q = Q_1 + Q_2$; $Q_1 = \{0,6,12,18,24\}$, $Q_2 = \{0,13\}$.

РОЗДІЛ 5
СПЕЦІАЛЬНИЙ КЛАС ПОЛІКОМБІНАТОРНИХ ЗАДАЧ ДИНАМІЧНОГО ПРОГРАМУВАННЯ

5.1. Оптимізація на загальній евклідовій полікомбінаторній множині методом динамічного програмування

Для ряду задач оптимізації на комбінаторних множинах процес розв'язування можна трактувати як багатокроковий. Це обумовлює доцільність застосування для таких типів задач методів динамічного програмування [136–140]. Відомий досвід застосування динамічного програмування в комбінаторній оптимізації (дивись, наприклад, [139]). До задач евклідової комбінаторної оптимізації метод динамічного програмування поки що не застосовувався.

Розглянемо клас задач евклідової комбінаторної оптимізації з лінійною цільовою функцією вигляду:

$$\min_{x \in E(G,\ H)} \sum_{i=1}^{n} \sum_{j=1}^{m_i} c_j x_{ij} \tag{5.1}$$

при обмеженнях

$$\sum_{i=1}^{n} \sum_{j=1}^{m_i} a_{ijk} x_{ij} \ \langle \ \Re \rangle \ \ b_k , \tag{5.2}$$

де $\Re \in \{\le, =, \ge\}$, $k = 1, 2, \dots, K$, $E(G,\ H)$ – евклідова полікомбінаторна множина, що являє собою декартовий добуток евклідових комбінаторних множин $E_1, E_2, \dots, E_n$:

$$E = E_1 \times E_2 \times \dots \times E_n .$$

Для цього класу задач вектор x представляється у вигляді:

$$x = \left(\underbrace{x_{11}, x_{12}, \dots, x_{1m_1}}_{x^1 \in E_1},\ \underbrace{x_{21}, x_{22}, \dots, x_{2m_2}}_{x^2 \in E_2}, \dots, \underbrace{x_{n1}, x_{n2}, \dots, x_{nm_n}}_{x^n \in E_n} \right).$$

Оптимальний розв'язок задач із даного класу може бути знайдений методом динамічного програмування. У цьому випадку метод динамічного програмування дозволяє перейти від задачі евклідової комбінаторної оптимізації великої вимірності до багатокрокової задачі, на кожному кроці розв'язування якої знаходиться оптимальний розв'язок задачі меншої вимірності.

Покажемо, що при цьому виконується необхідна умова застосування методу динамічного програмування [136] умова адитивності цільової функції (5.1):

$$\min_{x\in E(G,\,H)} \sum_{i=1}^{n}\sum_{j=1}^{m_i} c_j x_{ij} = \sum_{i=1}^{n} \min_{x\in E(G,\,H)} \left(\sum_{j=1}^{m_i} c_j x_{ij} \right). \quad (5.3)$$

Дійсно, площини $\sum_{j=1}^{m_i} c_j x_{ij} = 0$ при фіксованому значенні i мають той самий вектор нормалі $\bar{N} = \left(c_1, c_2, ..., c_{m_i}\right)$. Тому мінімум по $x \in E(G,\ H)$ при заданих обмеженнях (5.2) для усіх функцій $\sum_{j=1}^{m_i} c_j x_{ij}$ у (5.3) досягається в одній точці. Звідси безпосередньо випливає справедливість (5.3).

Умова адитивності функції (5.1) забезпечує розбиття процесу пошуку розв'язку задачі, тобто процесу переходу системи з початкового в кінцевий стан, на n етапів (за кількістю множин E_i). Перехід системи до стану $T^{(i)}$ здійснюється під дією керування u_i, що дає оптимальний розв'язок задачі мінімізації функції $\sum_{j=1}^{m_i} c_j x_{ij}$ на i-му етапі. При цьому очевидно, що перехід до стану $T^{(i)}$ не залежить від способу досягнення системою стану $T^{(i-1)}$, тобто розглянута задача задовольняє умові відсутності післядії.

Отже, метод динамічного програмування можна застосовувати до задач евклідової комбінаторної оптимізації виду (5.1), (5.2).

5.2. Застосування методу динамічного програмування до задачі розміщення об'єктів обслуговування

Розглянемо конкретизацію задачі (5.1), (5.2) для задачі розміщення об'єктів обслуговування, побудувавши алгоритм розв'язування цієї задачі в рамках методу динамічного програмування.

З точки зору динамічного програмування в даній задачі в якості системи виступає мережа об'єктів обслуговування, які розміщені в заданих населених пунктах. Під станом $T^{(k)}$ системи на k-му етапі ($k \in J_r$) будемо розуміти стан розвитку системи, під час якого проводиться побудова об'єктів обслуговування в одному з населених пунктів. Перехід до стану $T^{(k)}$ відбувається під дією керування u_k, що являє собою вибір кількості відділень визначеної ємності, що планується розмістити в населеному пункті.

Задача задовольняє наступні необхідні умови.

Умова відсутності післядії. Стан системи $T^{(k)}$ (обслуговування клієнтів в одному з населених пунктів) залежить від стану $T^{(k-1)}$ (тобто стан системи на k-му етапі залежить від того, які з населених пунктів вже обслуговані на попередніх

$(k-1)$ етапах) і обраного керування u_k (планування розміщення об'єктів з урахуванням вже обслугованих населених пунктів) і не залежить від того, яким способом система перейшла до стану $T^{(k-1)}$ (тобто, за скількома і якими етапами побудована мережа об'єктів обслуговування стану $T^{(k-1)}$).

Умова адитивності цільової функції. Очевидно, що мінімізація кількості клієнтів, які не обслуговуються (максимізація кількості клієнтів, що обслуговуються) у кожному населеному пункті дає загальну мінімальну кількість клієнтів, що не обслуговуються в усіх населених пунктах.

Отже, для функції (5.1) маємо:

$$\min_{z\in E}\sum_{j=1}^{s}\left(\sum_{i\in J_r, r_{ij}\le\rho}^{r} x_i^j-\sum_{i=1}^{\omega}y_i^j\right)=\sum_{j=1}^{s}\min_{z\in E}\left(\sum_{i\in J_r, r_{ij}\le\rho}^{r} x_i^j-\sum_{i=1}^{\omega}y_i^j\right).$$

Принцип оптимальності в термінах поставленої задачі буде виражатися в такий спосіб: яке б не було розміщення об'єктів обслуговування перед розміщенням об'єктів у черговому населеному пункті, необхідно так розмістити об'єкти для ще не обслугованих клієнтів, щоб сумарна кількість клієнтів, котрим уже надане обслуговування на цьому і наступних етапах забудови було максимальним.

Введемо позначення:

$F_{n-k}\left(T^{(k)}\right)$ – максимальна кількість клієнтів, що можна обслужити на останніх $(n-k)$ етапах (залежить від стана T^{k-1}); якщо кроків $n-k-1$, то маємо $F_{n-k-1}\left(T^{(k+1)}\right)$;

u_{k+1} – керування, що необхідно здійснити при переході від стану T^k до стану T^{k+1};

$W_{k+1}\left(T^{(k)},u_{k+1}\right)$ – максимальна кількість клієнтів, що може обслуговуватися на $(k+1)$-му етапі.

При цьому справедливе основне функціональне рівняння Беллмана:

$$F_{n-k}\left(T^{(k)}\right)=max[W_{k+1}\left(T^{(k)},u_{k+1}\right)+F_{n-k-1}\left(T^{(k+1)}\right)\ ],\quad k\in J_{n-1}^{0} \quad (5.4)$$

Таким чином, маємо наступний алгоритм розв'язування задачі.

Крок 1. Формуємо множину $X^s \subset E_{kv}(G)$ всіх допустимих векторів x^s, що можуть відповідати стану системи T^s на останньому s-му етапі. Кількість векторів x^s множини X^s позначимо через $\left|X^s\right| = \chi_s$.

Крок 2. Для векторів x^s із множини X^s визначаємо допустимий вектор $b^s \in E_{\eta n}^{\omega\omega}(Q,H)$, що задовольняє умовам (5.2) і

$$\sum_{i \in J_r,\ r_{ij} \le \rho} x_i^s - \sum_{j=1}^{\omega} b_j^s = \min_{t \in J_{\chi_s}} \left(\sum_{i \in J_r,\ r_{ij} \le \rho} x_i^s - \sum_{j=1}^{\omega} y_j^t \right). \quad (5.5)$$

Права частина рівності (5.5) являє собою значення цільової функції (5.1) на останньому s-му етапі. Позначимо його через $F(x^s)$, де $x^s \subset X^s$.

Множина тих x^s, для яких має місце (5.5), назвемо множиною оптимальних векторів (матриць) рівня T^s і позначимо X_*^s. Номер стану системи позначимо через k. На даному етапі $k = s$.

Крок 3. Якщо $k = 1$, те переходимо до кроку 6. Інакше – до кроку 5.

Крок 4. Розглянемо процес переходу від стану $T^{(k)}$ до стану $T^{(k-1)}$. Для кожної матриці з множини X_*^k визначаємо сукупність векторів $\tilde{x} \subset \tilde{X}^s \subset X^s$, елементи якої задовольняють умову:

$$\tilde{x}_i = a_i \text{ тільки тоді, коли } \sum_{j=k}^{s} x_i^j = 0. \quad (5.6)$$

Таким чином, утворюємо множину X^{k-1}, що містить матриці x^{k-1} вимірності $(s-k+2)\times k$. При цьому значення цільової функції $F(x^{k-1}) = F(x^k) + F(\tilde{x})$. Якщо $|\tilde{X}^s| = \varnothing$, то будемо вважати, що $\tilde{x}_i^s = 0$, $\forall i \in J_r$.

Крок 5. Визначаємо множину X_*^{k-1}, елементи якої x_*^{k-1} задовольняють умову:

$$F\left(x_*^{k-1}\right) = \min_{X^{k-1}} F\left(x^{k-1}\right). \quad (5.7)$$

Приймаємо $k = k - 1$. Переходимо до кроку 3.

Крок 6. Отримана сукупність матриць x_*^1 визначає оптимальне розміщення об'єктів обслуговування. При цьому відомо, комбінацією яких саме векторів $x^s \subset X^s$ представлений розв'язок задачі. А це означає, що відповідний набір визначених за умовою (5.5) векторів b^s дає оптимальний розподіл кількості відділень даних ємностей.

Розглянемо приклад розв'язування задачі із застосуванням наведеного алгоритму.

Нехай задана кількість населених пунктів $r = 4$. Відомо, що кількість клієнтів, що потребують обслуговування для кожного населеного пункту відповідно дорівнює $a_1 = 24$, $a_2 = 18$, $a_3 = 17$ і $a_4 = 10$ клієнтів. Задано радіус обслуговування $\rho = 15$ і матриця відстаней між населеними пунктами i і j:

$$r_{ij} = \begin{pmatrix} 0 & 10 & 22 & 25 \\ & 0 & 8 & 30 \\ & & 0 & 18 \\ & & & 0 \end{pmatrix}.$$

Дано, що в населених пунктах необхідно розмістити об'єкти двох типів $q = 7$, $q_2 = 11$ таким чином, щоб мінімізувати кількість клієнтів, що не зможуть обслуговуватися на даному рівні.

Кроки 1,2. $k = s = 4$. Формуємо X^4 – множину допустимих по радіусу обслуговування векторів і X_*^4 – множину оптимальних векторів стану T^4. (Наприклад, вектор (24, 18, 17, 0) є допустимим, оскільки $r_{12} = 10 < \rho$, $r_{23} = 8 < \rho$. Це означає, що в другому пункті можна побудувати об'єкт для спільного обслуговування клієнтів із I, II й III пунктів).

Таблиця 5.1. Множина допустимих векторів рівня T^4

№	X^4	Y^4	$F(x^4)$
1	(24, 0, 0, 0)	(0, 22)	2
2	(0, 18, 0, 0)	(7, 11)	0
3	(0, 0, 17, 0)	(14, 0)	3
4	(0, 0, 0, 10)	(7, 0)	3
5	(24,18, 0, 0)	(42, 0)	0
6	(24, 0, 17, 0)	(7, 33)	1
7	(0, 18, 17, 0)	(35, 0)	0
8	(24, 18, 17, 0)	(14, 44)	1

Таблиця 5.2. Множина оптимальних векторів рівня T^4

№	X_*^4
1	(0, 18, 0, 0)
2	(24, 18, 0, 0)
3	(0, 18, 17, 0)

Крок 3. Переходимо до кроку 5.

Крок 4. Розглянемо стан T^3. До *кожного* з трьох векторів із таблиці 5.2, для яких $F\left(x^4\right)=0$, множини X_*^4 приєднуємо вектори $\tilde{x}^4$ з таблиці 5.1, що задовольняють умові (5.6). Таким чином, визначимо матриці вимірності 2×4 – елементи множини X^3.

Таблиця 5.3. Множина допустимих матриць рівня T^3

№	X^3	$F(x^3)$
1	(0, 18, 0, 0) (24, 0, 0, 0)	2
2	(0, 18, 0, 0) (0, 0, 17, 0)	3
3	(0, 18, 0, 0) (0, 0, 0, 10)	3
4	(0, 18, 0, 0) (24, 0, 17, 0)	1
5	(24, 18, 0, 0) (0, 0, 17, 0)	3
6	(24, 18, 0, 0) (0, 0, 0, 10)	3
7	(0, 18, 17, 0) (24, 0, 0, 0)	2
8	(0, 18, 17, 0) (0, 0, 0, 10)	3

Крок 5. Визначаємо множину X_*^3 за умовою (5.7). Вона містить *єдину матрицю* x_*^3, оскільки мінімум функції $F(x_*^3)=1$ досягається лише на одній із матриць множини X^3 (рядок 4 із таблиці 5.3).

Приймаємо $k=3$ і переходимо до кроку 3.

Крок 3. $k=3\neq 1$. Отже, переходимо до кроку 5.

Кроки 4, 5. Розглянемо стан T^2. Для матриці x_*^3 визначаємо з умови (5.6) вектори $\tilde{x}^4$ з таблиці 5.1. Цій умові задовольняє лише вектор (0, 0, 0, 10). Таким чином, маємо (3×4)-мірну матрицю $x^2\equiv x_*^2$.

Таблиця 5.4. Множина допустимих і оптимальних матриць рівня T^2

$X^2 \equiv X^2_*$	$F(x^2) \equiv F(x^2_*)$
(0, 18, 0, 0) (24, 0, 17, 0) (0, 0, 0, 10)	4

Приймаємо $k = 2$ і переходимо до кроку 3.

Крок 3. $k = 2 \neq 1$. Отже, переходимо до кроку 5.

Кроки 4, 5. Розглянемо останній стан T^1. Для матриці x^2_* визначаємо з умови (5.6) вектори $\tilde{x}^4$ з таблиці 5.1. Таких векторів немає, отже, приймаємо $\tilde{x}^4_i = 0 \quad \forall i \in J_4$. Практично це означає, що на даному етапі забудова об'єктів обслуговування не відбувається. Таким чином, маємо (4×4)-мірну матрицю $x^1 \equiv x^1_*$.

Таблиця 5.5. Множина допустимих і оптимальних матриць рівня T^1

$X^1 \equiv X^1_*$	$F(x^1) \equiv F(x^1_*)$
(0, 18, 0, 0) (24, 0, 17, 0) (0, 0, 0, 10) (0, 0, 0, 0)	4

Приймаємо $k = 1$ і переходимо до кроку 3.

Крок 3. $k = 1$. Отже, переходимо до кроку 6.

Крок 6. Таким чином, оптимальне розміщення об'єктів обслуговування визначає множина $X^1_* = \{$ (0, 18, 0, 0), (24, 0, 17, 0), (0, 0, 0, 10), (0, 0, 0, 0)$\}$. Оскільки $r_{12} < \rho$, $r_{23} < \rho$, то клієнти з II пункту можуть обслуговуватися або в I, або в II, або в III пунктах. Отже, вектор (0, 18, 0, 0) може відповідати I, II й III пунктам. Аналогічно, вектор (24, 0, 17, 0), що характеризує розміщення об'єкта для спільного обслуговування клієнтів із I і II пунктів, може відповідати лише для II пункту, оскільки $r_{12} < \rho$, $r_{23} < \rho$, а $r_{13} > \rho$. Вектор (0, 0, 0, 10) відповідає лише IV пункту, оскільки $r_{i4} > \rho \quad \forall i \in J_3$. Таким чином, на базі цієї множини можна утворити три оптимальні розв'язки:

$$X^1_{*1} = (0, 18, 0, 0; 24, 0, 17, 0; 0, 0, 0, 0; 0, 0, 0, 10),$$

$$X^1_{*2} = (0, 0, 0, 0; 24, 18, 17, 0; 0, 0, 0, 0; 0, 0, 0, 10),$$

$$X^1_{*3} = (0, 0, 0, 0; 24, 0, 17, 0; 0, 18, 0, 0; 0, 0, 0, 10).$$

Значення цільової функції $F(x_*^1) = 4$. Отже, кількість клієнтів, що будуть обслуговуватися на вищому рівні при даному розміщенні об'єктів обслуговування дорівнює 4.

5.3. Порівняння різних методів розв'язування задачі розміщення об'єктів обслуговування

5.3.1. Аналіз програмної реалізації алгоритмів. Кожен із розглянутих методів в розділах 4 і 5 дає точний розв'язок задачі розміщення об'єктів обслуговування, але має свої переваги і недоліки.

Так, метод гілок і меж є найбільш універсальним, однак витрати програмних ресурсів при його реалізації на ЕОМ суттєво залежать від кількості елементів на рівнях галуження. Головною проблемою у ресурсному забезпеченні є розміщення в пам'яті ЕОМ елементів множини допустимих розв'язків задачі, які в програмі представляються у вигляді тривимірного масиву $X[t, \ i, \ j]$, де t – порядковий номер матриці (допустимого розв'язку), i – номер рядка матриці (рівня галуження), j – номер стовпця матриці (населеного пункту).

Сучасні мови програмування такі як об'єктно-орієнтована мова *Object Pascal* для середовища візуального програмування *Delphi* дозволяють використовувати сегмент даних обсягом до 2 Гб. Підрахуємо максимальну кількість допустимих розв'язків, яку можна представити в пам'яті ЕОМ для задачі розміщення об'єктів обслуговування клієнтів з n населених пунктів. Маємо масив цілочислових значень

X: array [1..t, 1..n, 1..n] of integer.

Кожна цілочислова змінна типу *integer* займає 4 байти пам'яті, а весь масив $t \times n \times n \times 4$ байт. Отже, $4n^2 t$ байт $\leq$ 2 Гб = $2 \cdot 2^{30}$ байт. Звідси, $t \leq \frac{2^{29}}{n^2}$. Наприклад, для 20 населених пунктів $t \leq \frac{2^{29}}{400} = 1342177 \approx 1{,}3$ млн. Але в програмі використовується велика кількість змінних та масивів, крім того, для запуску програми необхідна певна додаткова кількість вільної віртуальної пам'яті і тому практично верхній поріг значення t є значно меншим і становить величину порядку 500 тисяч.

Аналог методу ДальтонаЛлевеліна для задач евклідової комбінаторної оптимізації має найбільш просту постановку, оскільки дозволяє записати задачу (4.1), (4.2) безпосередньо у вигляді цільової функції та системи обмежень для програмної реалізації методу. Серед недоліків застосування цього методу в першу чергу слід зазначити стрімке зростання складності задачі (4.2), (4.10) – (4.12) внаслідок зростання числа змінних і обмежень задачі при збільшенні кількості пунктів обслуговування та кількості різних ємностей відділень.

Так, для n населених пунктів і ω різних величин об'єктів обслуговування кількість змінних задачі становить величину $n^2 + n \cdot \omega$, кожна з яких обирається із власних множин дискретних значень. Наприклад, для 20 населених пунктів і 6 різних величин об'єктів обслуговування маємо 520 змінних. Зведення задачі до канонічної форми ще значно збільшує кількість змінних і ставить під сумнів ефективність програмної реалізації даного методу, тому в роботі представлено програми розв'язування задачі розміщення об'єктів обслуговування за іншими алгоритмами.

Метод динамічного програмування має жорстко визначені рамки свого використання, що звужує коло задач, для яких він є прийнятним. Однак, як показано в пункті 5.1, існує широкий клас задач полікомбінаторної оптимізації з лінійною цільовою функцією, для яких метод динамічного програмування може бути застосований. Задача розміщення об'єктів обслуговування як задача евклідової комбінаторної оптимізації є представником такого класу задач.

Незначним недоліком цього методу можна вважати слабко прогнозовану зміну кількості допустимих матриць при переході до наступного рівня формування оптимального розв'язку задачі. Це змушує резервувати у програмі значний обсяг пам'яті для запобігання переповнення масиву допустимих матриць.

Суттєвою перевагою методу динамічного програмування є можливість розбиття процесу пошуку розв'язку задачі на етапи значно меншої складності. Поетапне формування оптимального розв'язку дозволило на відміну від методу гілок і меж використати для зберігання в пам'яті ЕОМ допустимих матриць двовимірні масиви даних, тим самим заощадивши програмні ресурси. Саме завдяки цьому реалізація алгоритму методу динамічного програмування дала найкращі результати за критерієм часу і витрат ресурсів оперативної пам'яті ЕОМ.

Як висновок аналізу результатів програмної реалізації різних методів розв'язування задачі розміщення об'єктів обслуговування слід зазначити, що основною характеристикою ефективності цих програм є обсяг ресурсів пам'яті ЕОМ, необхідних для знаходження оптимального розв'язку задачі. За цією характеристикою більш ефективною для розглядуваної задачі є програмна реалізація методу динамічного програмування.

При аналізі результатів також необхідно враховувати, що максимально допустимий обсяг ресурсів суттєво залежить від типу процесора, обсягу оперативної і віртуальної пам'яті та інших показників апаратної частини ЕОМ.

5.3.2. Результати числових експериментів. Приклади розрахунків за методами гілок і меж та динамічного програмування за програмою, написаною мовою *Object Pascal* в середовищі візуального програмування *Delphi* (див. додаток А) на ЕОМ із процесором *Celeron* 1.2 GHz і оперативною пам'яттю 256 Мбайт, наведені в таблиці 5.6.

Таблиця 5.6. Результати числових експериментів за методами гілок і меж та динамічного програмування

№	s	ω	$\sum_{i=1}^{s} a_i$	m	n^1_{max}	n^2_{max}	n_y	t_1	t_2
1	5	3	61	61	768	28	30	00:00:01	00:00:01
2	6	4	139	139	9000	63	4212	00:00:02	00:00:01
3	6	5	147	147	1152	207	79200	00:00:06	00:00:01
4	7	3	81	80	10800	770	336	00:00:02	00:00:01
5	10	5	173	172	311040	1560	4320	00:02:59	00:00:02
6	10	6	174	171	311040	54198	201600	00:06:14	00:00:02
7	15	5	304	275	259200	2290	259020	00:10:05	00:00:02
8	20	4	447	426	82944	102960	194832	00:11:50	00:00:08
9	20	6	422	398	442368	176214	280896	01:00:28	00:00:35
10	20	10	451	409	442368	20736	3491136	–	00:00:47

s – кількість населених пунктів;

ω – кількість різних ємностей відділень;

$\sum_{i=1}^{s} a_i$ – кількість клієнтів у всіх населених пунктах;

m – кількість обслугованих клієнтів;

n^1_{max} – максимальна кількість допустимих матриць за методом гілок і меж;

n^2_{max} – максимальна кількість допустимих матриць за методом динамічного програмування;

n_y – кількість допустимих векторів y^j ;

t – час роботи програми за методом гілок і меж в форматі год:хв:сек;

t_2 – час роботи програми за методом динамічного програмування в форматі год:хв:сек.

Останній приклад з таблиці 5.6 для методу гілок і меж викликає помилку часу виконання програми внаслідок переповнення сегменту даних.

Дані ρ, a_i, q_i, r_{ij} $\forall i,j \in J_s$ $(i \neq j)$ для розрахунків сформовані випадковим чином за допомогою функції *Random* – генератора псевдорівномірно розподілених випадкових чисел мови *Object Pascal* в інтервалах від 0 до 50 (для a_i и q_i) і від 0 до 30 (для r_{ij} і ρ); s і ω – задаються.

Детальний вивід результатів роботи програми для деяких задач з таблиці 5.6 представлений у додатку Б (метод гілок і меж) та у додатку В (метод динамічного програмування).

5.3.3. Аналіз функціональної залежності за допомогою пакету «Експерт кривих». Використовуючи пакет «Експерт кривих» (*Curve Expert*) [176], можна оцінити залежність часу виконання програми від початкових параметрів задачі, що визначають її вимірність. Для цього за даними з таблиці 5.6 вважаємо аргументом загальну кількість різних ємностей відділень в усіх населених пунктах (величину $s \cdot \omega$), а залежною змінною – час роботи програми за певним методом (t або t_2).

Регересійний аналіз результатів, отриманих за методом гілок і меж, представлений на рис. 5.1, 5.2, а за методом динамічного програмування – на рис. 5.3, 5.4.

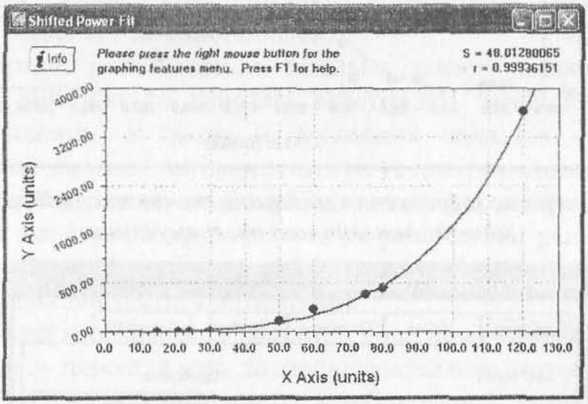

Рис. 5.1. Графічне зображення множини точок та лінії регресії для методу гілок і меж

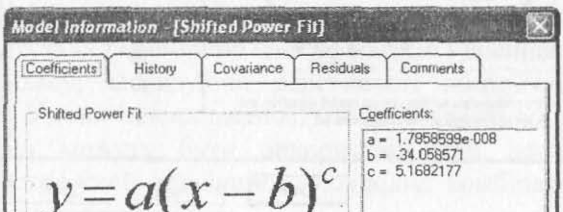

Рис. 5.2. Аналітичний вираз лінії регресії для методу гілок і меж

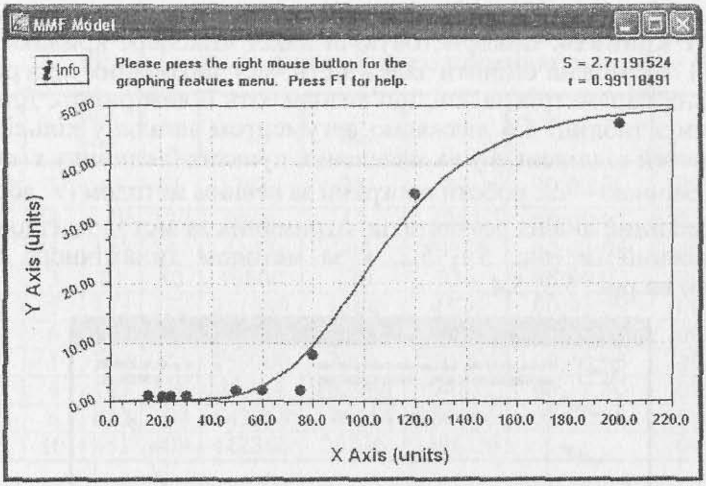

Рис. 5.3. Графічне зображення множини точок та лінії регресії для методу динамічного програмування

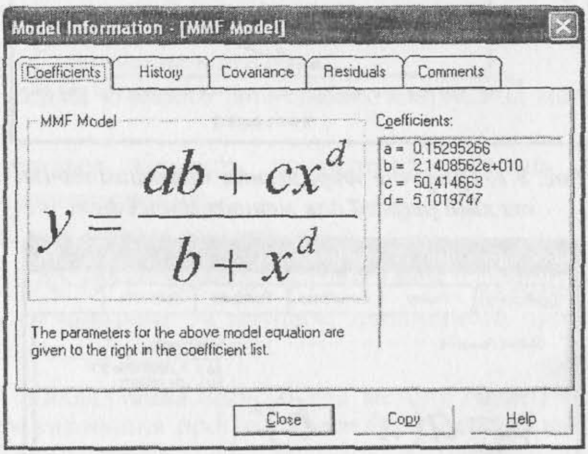

Рис. 5.4. Аналітичний вираз лінії регресії для методу динамічного програмування

Аналітичні вирази для кривих регресій, що представлені на рис. 5.2 і 5.4, виражають залежність часу виконання від вимірності задачі. Їх вигляд вселяє надію, що програмна реалізація методів є практично ефективною.

Актуальною для дослідження залишається задача теоретичної оцінки трудомісткості запропонованих методів.

ПІСЛЯМОВА

Евклідова комбінаторна оптимізація є прогресивним та перспективним напрямом досліджень в області математичного програмування та проблем дослідження операцій, пов'язаних з інформатикою. Серед комбінаторних об'єктів, які потребують всебічного дослідження важливе місце займають більш складні структури – полікомбінаторні множини. Інтерес до дослідження апарату полікомбінаторних множин викликаний їх великим практичним значенням щодо постановки та розв'язування задач комбінаторної оптимізації, які описують економічні, соціальні та інші процеси.

У монографії розглядається загальна задача полікомбінаторної оптимізації, виявлені та доведені властивості загальної полікомбінаторної множини, а також її частинних випадків – евклідових комбінаторних множин поліпереставлень та полірозміщень.

Методи дослідження полікомбінаторних множин, що використовуються в монографії, ґрунтуються на розвинутих раніше теорії та методах евклідової комбінаторної оптимізації, а також на інших відомих методах дискретної оптимізації.

Встановлення необхідних та достатніх умов невиродженості загальних многогранників переставлень та поліпереставлень дозволяє застосовувати для їх дослідження апарат комбінаторної теорії многогранників, численні методи якого стосуються, в основному, саме невироджених многогранників. Доведення достатньої умови еквівалентності комбінаторних многогранників одного типу дозволяє визначити класи еквівалентності за їх первинною специфікацією і досліджувати спільні властивості класу, аналізуючи властивості лише окремого його представника. Умови невиродженості та еквівалентності переставних многогранників можуть бути використані при дослідженні цих важливих властивостей для інших евклідових комбінаторних і полікомбінаторних многогранників.

Головне значення отримання незвідної системи для загальних множин розміщень і полірозміщень полягає, на наш погляд, у тому, що кожне з обмежень незвідної системи описує гіпергрань, що є частиною опуклої оболонки відповідного комбінаторного многогранника. Відсутність надлишкових обмежень дозволяє значно спростити задачі, що розв'язуються, зменшивши кількість досліджуваних обмежень. Це має велике практичне значення і може бути використано при програмній реалізації задач на розміщеннях для підвищення ефективності відповідних алгоритмів та економії ресурсів ЕОМ.

Отриманий точний розв'язок задачі розміщення об'єктів обслуговування доводить перевагу математичної моделі цієї задачі як задачі евклідової полікомбінаторної оптимізації в порівнянні з існуючим розв'язком, що здійснюється евристичним алгоритмом. Використання різних методів забезпечує гнучкість вибору алгоритму для отримання розв'язку з урахуванням конкретної специфіки певної задачі. Побудовані алгоритми та їх програмна реалізація можуть бути застосовані в реальних задачах планування розміщення об'єктів виробництва або сфери обслуговування.

Виокремлення спеціального класу задач на загальній евклідовій полікомбінаторній множині забезпечує виконання необхідних умов відсутності післядії та адитивності цільової функції. Це дозволило вперше застосувати метод динамічного програмування для отримання розв'язку певних задач евклідової комбінаторної оптимізації. Ефективність такого підходу для задачі розміщення об'єктів обслуговування як представника цього класу, підтверджена чисельними експериментами. Властивості спеціального класу задач дозволяють представляти процес розв'язування оптимізаційних задач як багатокроковий і використовувати метод динамічного програмування для отримання розв'язку задач оптимізації на інших полікомбінаторних множинах.

ЛІТЕРАТУРА

1. Сергиенко И.В., Шило В.П. Задачи дискретной оптимизации: проблемы, методы исследования, решения. – К.: Наукова думка, 2003. – 263 с.
2. Сергиенко И.В., Каспшицкая М.Ф. Модели и методы решения на ЭВМ комбинаторных задач оптимизации. – К.: Наукова думка, 1981. – 288 с.
3. Михалевич В.С., Сергиенко И.В., Шор Н.З. Исследование методов решения оптимизационных задач и их приложений // Кибернетика. – 1981. – № 4. – С. 89–113.
4. Сергиенко И.В. Математические модели и методы решения задач дискретной оптимизации. – К.: Наук. думка, 1988. – 472 с.
5. Козерацкая Л.Н., Лебедева Т.Т., Сергиенко И.В. Исследование устойчивости задач дискретной оптимизации // Кибернетика и системный анализ. – 1993. – № 3. – С. 78–93.
6. Сергиенко И.В. О некоторых направлениях и результатах работ в области математического программирования и системного анализа // Кибернетика и системный анализ. – 1995.– № 3. – С. 3–48.
7. Сергиенко И.В., Козерацкая Л.Н., Лебедева Т.Т. Исследование устойчивости и параметрический анализ дискретных оптимизационных задач. – К.: Наук. думка, 1995. – 171 с.
8. Козерацкая Л.Н., Лебедева Т.Т., Сергиенко И.В. Задачи дискретной оптимизации: исследование устойчивости // Обозрение прикладной и промышленной математики. – 1995. – № 1. – С. 12–30.
9. Сергиенко И.В., Козерацкая Л.Н., Кононова А.А. Устойчивость и неограниченность задач векторной оптимизации // Кибернетика и системный анализ. – 1997. – № 1. – С. 3–10.
10. Сергиенко И.В., Михалевич М.В., Стецюк П.И., Кошлай Л.Б. Межотраслевая модель планирования структурно-технологических изменений // Кибернетика и системный анализ. – 1998. – № 3. – С. 3–17.
11. Сергієнко І.В. Інформатика в Україні: становлення, розвиток, проблеми. – К.: Наук. думка, 1999. – 354 с.
12. Сергиенко И.В., Шило В.П., Рощин В.А. РЕСТАРТ-технология решения задач дискретной оптимизации // Кибернетика и системный анализ. – 2000. – № 5. – С. 32–40.
13. Сергиенко И.В., Лебедева Т.Т., Семенова Н.В. О существовании решений в задачах векторной оптимизации // Кибернетика и системный анализ. – 2000. – № 6. – С. 39–46.
14. Гуляницкий Л.Ф., Кошлай Л.Б., Сергиенко И.В. О сходимости одного метода вероятностного моделирования для решения задач комбинаторной оптимизации // Кибернетика и системный анализ. – 1993. – № 3. – С. 164–166.

15. Колоколов А.А. Регулярные разбиения и отсечения в целочисленном программировании // Сиб. журн. исследования операций. – 1994. – № 2. – С. 18–39.
16. Kolokolov A.A. On the L-structure of integer linear programming problems // System Modelling and Optimization. Proceedings of the 16th IFIP conference on the modelling and optimization.-Springer Verlag, 1993. – P. 756–760.
17. Kolokolov A.A. Decomposition Algorithms for Solving of some Production-Transportation Problems // Triennal Symposium on Transportation Analysis II. Preprints. Vol.1. – Capri, Italy, 1994. – P. 179–183.
18. Kolokolov A.A. Some L-class enumeration algorithms for integer programming problems // Proceedings of the 3rd IFIP WG-7.6 Working Conference on Optimization-Based Computer-Aided Modelling and Design, Published by IITA, Prague, Czech Republic, 1995. – P. 256–260.
19. Киселева Е.М., Шор Н.З. Исследование алгоритмов решения одного класса непрерывных задач разбиения // Кибернетика и системный анализ. – 1994. – № 1. – С. 84–96.
20. Стоян Ю.Г. Некоторые свойства специальных комбинаторных множеств. – Харьков, – 1980. – 22 с. (Препринт / АН УССР. Ин-т пробл. машиностроения; 85).
21. Стоян Ю.Г. Об одном отображении комбинаторных множеств в евклидово пространство. – Харьков, – 1982. – 33 с. (Препринт / АН УССР. Ин-т пробл. машиностроения; 173).
22. Стоян Ю.Г., Яковлев С.В. Математические модели и оптимизационные методы геометрического проектирования. – К.: Наук. думка, 1986. – 268 с.
23. Стоян Ю.Г., Соколовский В.З. Решение некоторых многоэкстремальных задач методом сужающихся окрестностей. – К.: Наукова думка, 1988. – 268 с.
24. Стоян Ю.Г., Ємець О.О. Теорія і методи евклідової комбінаторної оптимізації. – К.: Інститут системних досліджень освіти, 1993. – 188с.
25. Стоян Ю.Г., Романова Т.Е., Евсеева Л.Г. Комбинаторная оптимизационная задача размещения прямоугольников с учетом погрешности исходных данных // Доповіді НАН України. – 1997. – № 7. – С. 56–60
26. Стоян Ю.Г., Яковлєв С.В., Емец О.А., Валуйская О.А. Построение выпуклых продолжений для функций заданых на гиперсфере // Кибернетика и системный анализ. – 1998. – № 2. – С. 1–11.

27. Стоян Ю.Г., Яковлєв С.В., Ємець О.О., Валуйська О.О. Про існування опуклого продовження функцій, які задані на гіперсфері // Доповіді НАН України. – 1998. – № 2. – С. 128–133.
28. Стоян Ю.Г., Ємець О.О., Ємець Є.М. Множини полірозміщень в комбінаторній оптимізації // Доповіді НАН України. – 1999. – № 8. – С. 37–41.
29. Стоян Ю.Г., Ємець О.О., Ємець Є.М. Оптимізація на полірозміщеннях: теорія та методи: Монографія. – Полтава, РВЦ ПУСКУ, 2005. – 103 с.
30. Яковлев С.В. Теория и оптимизационные методы геометрического проектирования в системах управления, наблюдения и контроля: Автореф. дис. ... д-ра. физ.-мат. наук. – Томск, 1989. – 27 с.
31. Яковлев С.В., Гребенник И.В. О некоторых классах задач оптимизации на множестве размещений и их свойствах // Известия вузов. Математика. – 1991. – № 11. – С. 74–86.
32. Яковлев С.В., Гребенник И.В. Локализация решений некоторых нелинейных целочисленных задач оптимизации // Кибернетика и системный анализ. – 1993. – № 5. – С. 116–123.
33. Яковлев С.В., Валуйская О.А. О минимизации линейной функции на вершинах перестановочного многогранника с учетом линейных ограничений // Доповіді НАН України. – 1999. – № 4. – С. 103–108.
34. Яковлев С.В., Валуйская О.А. О минимизации линейной функции на вершинах перестановочного многогранника с учетом линейных ограничений // Укр. мат. журн. – 2001. – Т. 53. – № 9. – С. 89–101.
35. Козерацька Л.М. Цілочислові задачі оптимізації: проблема стійкості і параметричний аналіз: Автореф. дис. ... д-ра. фіз.-мат. наук: 01.05.01 / Інститут кібернетики ім. В.М. Глушкова НАН України. – К., 1997. – 32 с.
36. Козерацкая Л.Н. Множество строго эффективных точек задачи частично целочисленной векторной оптимизации как характеристика ее устойчивости // Кибернетика и системный анализ. – 1997. – № 6. – С. 181–184.
37. Емеличев В.А., Комлик В.И. Применение динамического программирования к решению задач размещения // Докл. АН БССР. – 1966. – Т. 10. – С. 721–725.
38. Емеличев В.А., Ковалев М.М., Кравцов М.К. Многогранники, графы, оптимизация. – М.: Наука, 1981. – 344 с.
39. Емеличев В.А., Бердышева Р.А. О радиусах устойчивости, квазиустойчивости и стабильности векторной траекторной задачи

лексикографической оптимизации // Дискретная математика. – 1998. – Т. 10, вып. 1. – С. 20–27.
40. Емеличев В.А., Бердышева Р.А. Об условиях устойчивости векторной траекторной задачи лексикографической дискретной оптимизации // Кибернетика и системный анализ. – 1998. – № 6. – С. 120–127.
41. Пшеничный Б.Н., Ненахов Э.И., Кузьменко В.Н. Модифицированный метод отсечения для минимизации выпуклой функции // Кибернетика и системный анализ. – 1997. – № 6. – С. 142–149.
42. Пшеничный Б.Н., Ненахов Э.И., Кузьменко В.Н. Комбинаторный метод решения общей задачи выпуклого программирования // Кибернетика и системный анализ. – 1998. – № 4. – С. 121–134.
43. Линейное и нелинейное программирование / И.Н. Ляшенко, Е.А. Карагодова, Н.В. Черникова, Н.З. Шор – К.: Вища школа, 1975. – 372 с.
44. Павлов А.А., Павлова Л.А. Основы методологии проектирования ПДС-алгоритмов для труднорешаемых комбинаторных задач // Проблемы информатики и управления. – 1995. – № 4. – С. 135–141.
45. Павлов А.А., Ван Инхуэйн. Особенности решения NP – трудных задач комбинаторной оптимизации // Інформатика та нові технології.– 1997. – № 1. – С. 13.
46. Павлов О.А., Павлова Л.О. Принцип розпаралелювання обчислень як засіб підвищення ефективності ПДС-алгоритмів для важкорозв'язуваних комбінаторних задач // Наукові вісті НТУУ «КПІ». – 1997. – № 1 – С. 22–26.
47. Павлова Л.О. Розробка та дослідження ПДС-алгоритмів для важкорозв'язуваних задач комбінаторної оптимізації: Автореф. дис. ... канд. фіз.-мат. наук: 01.05.01 / Інститут кібернетики ім. В.М.Глушкова НАН України. – К., 1996. – 19 с.
48. Асельдеров З.М., Павлов А.А., Павлова Л.А. Оценка эффективности ПДС-алгоритмов и статистическое моделирование // Мат. машины и системы. – 1998. – № 1. – С. 52–56.
49. Гордеев Э.Н. Алгоритмический и постоптимальный анализ устойчивости решений задач дискретной оптимизации: Дис. ... д-ра физ.-мат. наук. 1992. – 247 с.
50. Гордеев Э.Н., Леонтьев В.К. Общий подход к исследованию устойчивости задач дискретной оптимизации // Журн. вычисл. математики и мат. физики. – 1996. – Т. 36, № 1. – С. 66–72.
51. Гордеев Э.Н. Исследование устойчивости в оптимизационных задачах на матроидах в метрике l1 // Кибернетика и системный анализ. – 2001. – № 2. – С. 132–144.

52. Кісельова О.М., Гарт Л.Л., Фірсов О.Д. Генетичні алгоритми розв'язання оптимізаційних задач та їх застосування: Навч. посіб. Дніпропетр. нац. ун-т. – Дніпропетровськ, 2001. – 48 с.
53. Ковалев М.М. Дискретная оптимизация. – Минск: Изд-во БГУ, 1977. – 192 с.
54. Ковалев М.М., Исаченко А.Н., Нгуен Нгиа. Линеаризация комбинаторных задач оптимизации // Доклады АН БССР. – 1978. – Т. 22. – № 10. – С. 869–872.
55. Емец О.А. Свойства специальных комбинаторных задач оптимизации, методы и алгоритмы их решения: Автореф. дис. ... канд. физ.-мат. наук: 01.01.09 / ВЦ АН СССР. М., – 1985. – 16 с.
56. Емец О.А. Свойства специальных комбинаторных задач оптимизации, методы и алгоритмы их решения: Дис. ... канд. физ.-мат. наук: 01.01.09. – Харьков: ХИРЭ, 1985. – 155 с.
57. Емец О.А. Задачи оптимизации на евклидовом полиперестановочном множестве с повторениями: свойства допустимого множества // Методы и программные средства оптимизации, моделирования и создания вычислительных систем: Сб. научн. тр. – К.: АН УССР. Ин-т кибернетики им. В.М. Глушкова. 1990. – С. 22–24.
58. Емец О.А. Евклидовы комбинаторные множества и оптимизация на них. Новое в математическом программировании: Учеб. пособие. – К.: УМК ВО. – 1992. – 92 с.
59. Емец О.А. Комбинаторная модель и приближенный метод с априорной оценкой решения оптимизационной задачи размещения разноцветных прямоугольников // Экономика и матем. Методы – 1993. – Т. 29, вып. 2. – С. 294–304.
60. Емец О.А. Об экстремальных свойствах недифференцируемых выпуклых функций на евклидовом множестве сочетаний с повторениями // Укр.мат.журн. – 1994. – 46, № 6. – С. 680–691.
61. Емец О.А. Об оптимизации выпуклых функций на евклидовом комбинаторном множестве полиперестановок // Журн. вычислит. математики и мат. физики. – 1994. – № 6. – С. 855–869.
62. Yemets O.A. Extremal properties of nondifferentiable convex functions on euclidean sets of combinations with repetitions // Ukrainian Mathematical Journal. – 1994. – V. 46, № 6. – P. 735–747.
63. Yemets O.A. The optimization of linear and convex functions on a Euclidean combinatorial set of polypermutations // Comp. Maths. Math. Phys. – 1994. – V. 34, № 6. – P. 737–748.
64. Емец О.А. Об одном методе отсечения для задач комбинаторной оптимизации // Экономика и мат. методы. – 1997. – Т. 33, № 4. – С. 46–51.

65. Ємець О.О., Ємець Є.М. Метод відсікання в евклідовій комбінаторній оптимізації: Навч. посіб. – Полтава, 1997. – 30 с.
66. Ємець О.О. Теорія і методи комбінаторної оптимізації на евклідових множинах в геометричному проектуванні: Автореф. дис. ... д-ра фіз.-мат. наук: 01.05.01 / Ін-т кібернетики НАН України. – К., 1997. – 42 с.
67. Ємець О.О. Теорія і методи комбінаторної оптимізації на евклідових множинах в геометричному проектуванні: Дис. ... д-ра фіз.-мат. наук: 01.05.01. – Харків: ХДТУРЕ, 1997. – 242 с.
68. Euclidean combinatorial sets in combinatorial optimization / Yemets O.; Полт. держ. техн. ун-т ім. Юрія Кондратюка. – Полтава, 1998. – 11 с. – Деп. в ДНТБ України 10.09.98, № 397. – Ук 98.
69. Емец О.А., Емец Е.М., Колечкина Л.Н. Использование метода отсечений при раскрое // Радиоэлектроника и информатика. – 1998. – № 3. – С. 114–117.
70. Ємець О.О., Роскладка А.А. До оптимізації оцінок екстремальних значень сильноопуклих функцій при оптимізації на сполученнях // Вісник державного університету «Львівська політехніка». – 1998. – № 337. – С. 320–322.
71. Ємець О.О., Ємець Є.М. Використання моделей і методів комбінаторної оптимізації в задачах економії ресурсів // Проблеми праці, економіки та моделювання: Зб наукових праць. Ч. 1. / Міносвіти України, Технолог. ун-т Поділля. – Хмельницький: НВП "Еврiка", 1998. – С. 92–93.
72. Ємець О.О., Роскладка А.А. Параметрична комбінаторна оптимізація на сполученнях // Вісник державного університету «Львівська політехніка». – 1998. – № 364. – С. 32–34.
73. Ємець О.О., Недобачій С.І. Загальний переставний многогранник: незвідна система лінійних обмежень та рівняння всіх гіперграней // Наукові вісті НТУУ «КПІ». – 1998. – № 1. – С. 100–106.
74. Ємець О.О., Роскладка А.А. Побудова опуклої оболонки загальної множини сполучень // Радиоэлектроника и информатика. – 1998. – № 4. – С. 95–96.
75. Ємець О.О., Колєчкіна Л.М. Розв'язування оптимізаційних задач з дробово-лінійною цільовою функцією на загальній множині переставлень // Вісник держ. ун-ту «Львівська політехніка». – 1998. – № 337, «Прикладна математика». – Т. 2. – С. 317–320.
76. Ємець О.О., Колєчкіна Л.М., Недобачій С.І. Дослідження областей визначення задач евклідової комбінаторної оптимізації на переставних множинах. Ч. 1. – Полтава: ЧПКП «Легат», 1999. – 64 с.

77. Ємець О.О., Колєчкіна Л.М., Недобачій С.І. Дослідження областей визначення задач евклідової комбінаторної оптимізації на переставних множинах. Ч. 2. Про одну задачу оптимізації на переставленнях. – Полтава: ЧПКП «Легат», 1999. – 32 с.
78. Емец О.А., Роскладка А.А. Алгоритмическое решение двух параметрических задач оптимизации на множестве сочетаний с повторениями // Кибернетика и системный анализ. – 1999. – № 6. – С. 160–165.
79. Yemets O.A., Roskladka A.A. On estimates of minima criterion functions in optimization on combinations // Ukrainian Mathematical Journal. –1999. – Vol. 51. – № 8. – P. 1262–1265.
80. Емец О.А., Роскладка А.А. Модель выбора подмножества задач для ЭВМ с минимизацией количества периферийных устройств как задача комбинаторной оптимизации // Проблемы бионики. – 1999. – № 51. – С. 158–161.
81. Ємець О.О., Роскладка А.А. Про оцінки мінімумів цільових функцій при оптимізації на сполученнях // Укр. мат. журн. – 1999. – Т. 51. – № 8. – С. 1118–1121.
82. Ємець О.О., Чілікіна Т.В. Нелінійні задачі комбінаторної оптимізації на вершинно-розташованих множинах та методи їх роз'язування // Динамические системы. – 2004. – Вып. 17. – Симферополь: Тавр. нац. ун-т. – С. 205–209.
83. Емец О.А., Роскладка А.А., Роскладка Е.В. Применение евклидовых поликомбинаторных множеств к построению моделей оптимизационных задач // Abstracts of second international school on actuarial and financial mathematics (June 8-12, 1999, Kyiv, Ukraine). – C. 20.
84. Yemets O.A., Kolechkina L.M. Optimization problem on permutations with linear-fractional objective function: properties of the set of admissible solutions // Ukrainian Mathematical Journal. – 2000. – V. 52. № 12. – P. 1858–1871.
85. Емец О.А., Емец Е.М. Моделирование некоторых инвестиционных задач с помощью евклидовой комбинаторной оптимизации // Экономика и матем. методы. – 2000. – Т. 36. – № 2. – С. 141–144.
86. Ємець О.О., Ємець Є.М. Відсікання в лінійних частково комбінаторних задачах евклідової комбінаторної оптимізації // Доповіді НАН України. – 2000. – № 9. – С. 105–109.
87. Ємець О.О., Ємець Є.М. Оцінки та достатні умови мінімуму сильно опуклої функції при її мінімізації на розміщеннях // Волинський математичний вісник. – 2000. – № 7. – С. 67–69.
88. Ємець О.О., Колєчкіна Л.М. Задача оптимізації на переставленнях з дробово-лінійною цільовою функцією: властивості множини

допустимих розв'язків // Укр. мат. журн. – 2000. – Т. 52. – № 12. – С. 1630–1640.
89. Ємець О.О., Колєчкіна Л.М. Моделювання деяких прикладних задач оптимізаційними задачами з дробово-лінійною функцією цілі на переставленнях // Волинський математичний вісник. – 2000. – № 7. – С. 70–77.
90. Roskladka O.V., Yemets O.O., Nedobachiy S.I. About the system of linear restrictions, which describe a general polyhedron of the arrangements // VIII міжнародна наук. конф. ім. ак. М.Кравчука (11–14 травня 2000 р., Київ): Мат-ли конф. – К., – 2000. – С. 354.
91. Емец О.А., Емец Е.М. Отсечения в линейных частично комбинаторных задачах оптимизации на перестановках // Экономика и матем. методы. – 2001. – Т. 37. – № 1. – С. 118–121.
92. Емец О.А., Евсеева Л.Г., Романова Н.Г. Интервальная математическая модель комбинаторной задачи цветной упаковки прямоугольников // Кибернетика и системный анализ. – 2001. – № 3. – С. 131–138.
93. Емец О.А., Недобачий С.И., Колечкина Л.Н. Неприводимая система ограничений комбинаторного многогранника в дробно-линейной задаче оптимизации на перестановках // Дискретная математика. – 2001. – Т. 13. – Вып. 1. – С. 110–118.
94. Емец О.А., Барболина Т.Н. Решение линейных задач оптимизации на размещениях методом отсечений // Кибернетика и системный анализ. – 2003. – № 6. – С. 131–141.
95. Емец О.А., Барболина Т.Н. Решение задач комбинаторной оптимизации методом построения лексикографической эквивалентности // Кибернетика и системный анализ. – 2004. – № 5. – С. 115–125.
96. Емец О.А., Колечкина Л.Н. Задачі комбінаторної оптимізації з дробово-лінійними цільовими функціями: Монография. – К.: Наукова думка, 2005. – 117 с.
97. Емец О.А., Роскладка Е.В. Многоуровневая задача обслуживания как задача евклидовой комбинаторной оптимизации и ее решение // Динамические системы. – 2001. – Вып. 17. – С. 205–209.
98. Valuiskaya O.A., Yemets O.A., Romanova N.G. Stoyan-Yakovlev's modified method applied to convex continuation of polynomials defined on polypermutations // Computational Mathematics and Mathematical Physics. – 2002. – V. 42. – №. 42. – P. 566–570.
99. Валуйська О.О., Ємець О.О., Пічугіна О.С. До питання про нелінійну та параметричну оптимізацію на комбінаторних множинах // Вісник Львів. нац. ун-ту. Сер. приклад. математика та інформатика. – 2002. – Вип. 4. – С. 94–101.

100. Валуйская О.А., Емец О.А., Романова Н.Г. Выпуклое продолжение многочленов, заданных на полиперестановках, модифицированным методом Стояна-Яковлева // Журн. вычислит. математ. и матем. физики. – 2002. – Т. 42. – № 4. – С. 591–596.
101. Емец О.А., Роскладка Е.В. Решение некоторых евклидовых комбинаторных задач оптимизации методом динамического программирования // Кибернетика и системный анализ. – 2002. – № 1. – С. 138–146.
102. Ємець О.О., Романова Н.Г., Роскладка О.В. Про властивості деяких задач евклідової комбінаторної оптимізації на переставленнях та методи їх розв'язування // Вісник Львів. нац. ун-ту. Сер. приклад. математика та інформатика. – 2002. – Вип. 4. – С. 94–101.
103. Ємець О.О., Роскладка О.В., Недобачій С.І. Незвідна система обмежень для загального многогранника розміщень // Укр. мат. журн. – 2003. – Т. 55. – № 1. – С. 3–11.
104. Євсеєва Л.Г. Комбінаторна оптимізаційна задача розміщення прямокутників та методи її розв'язання з урахуванням похибок початкових даних: Автореф. дис. ... канд. фіз.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України. – Харків, 1995. – 20 с.
105. Пичугина О.С. Методы и алгоритмы решения некоторых задач оптимизации на множествах сочетаний и размещений: Автореф. дис. ... канд. физ.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України. – Харьков., 1996 – 24 с.
106. Пичугина О.С. Методы и алгоритмы решения некоторых задач оптимизации на множествах сочетаний и размещений: Дис. ... канд. физ.-мат. наук: 01.05.02. – Харьков, 1996. – 169 с.
107. Валуйська О.О. Метод опуклого продовження для моделювання задач оптимізації геометричного проектування: Автореф. дис. ... канд. фіз.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України. – Харків, 1996. – 24 с.
108. Валуйська О.О. Метод опуклого продовження для моделювання задач оптимізації геометричного проектування: Дис. ... канд. физ.-мат. наук: 01.05.02. – Харків: ХДТУРЕ, 1996. – 116 с.
109. Недобачій С.І. Моделі, методи і алгоритми в задачах евклідової комбінаторної оптимізації: Автореф. дис.... канд. фіз.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України. – Харків, 1999. – 17 с.
110. Недобачій С.І. Моделі, методи і алгоритми в задачах евклідової комбінаторної оптимізації: Дис. ... канд. физ.-мат. наук: 01.05.02. – Полтава: ПДТУ, 1999. – 143 с.

111. Роскладка А.А. Параметричні задачі та стійкість при моделюванні евклідовими комбінаторними задачами оптимізації: Автореф. дис.... канд. фіз.-мат. наук: *01.05.01* / Дніпр. нац. ун-т. – Дніпропетровськ, 2000. – 18 с.
112. Роскладка А.А. Параметричні задачі та стійкість при моделюванні евклідовими комбінаторними задачами оптимізації: Дис. ... канд. фіз.-мат. наук: 01.05.01. – Полтава: ПДТУ, 2000. – 142 с.
113. Колєчкіна Л.М. Властивості задач комбінаторної оптимізації з дробово-лінійними цільовими функціями. Методи та алгоритми їх розв'язання: Автореф. дис. ... канд. фіз.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України.– Харків, 2002. – 19 с.
114. Колєчкіна Л.М. Властивості задач комбінаторної оптимізації з дробово-лінійними цільовими функціями. Методи та алгоритми їх розв'язання: Дис. ... канд. физ.-мат. наук: 01.05.02. – Полтава: ПНТУ, 2002. – 124 с.
115. Ємець Є.М. Дослідження властивостей математичних моделей комбінаторних задач оптимізації на полірозміщеннях та розробка методу і алгоритму комбінаторного відсікання: Автореф. дис... канд. фіз.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України. – Харків, 2002. – 19 с.
116. Ємець Є.М. Дослідження властивостей математичних моделей комбінаторних задач оптимізації на полірозміщеннях та розробка методу і алгоритму комбінаторного відсікання: Дис. ... канд. физ.-мат. наук: 01.05.02. – Харьков, 2002. – 155 с.
117. Роскладка О.В. Незвідна система обмежень загального многогранника полірозміщень // Вісник Запорізького державного університету. – 2002. – № 2. – С. 81–84.
118. Roskladka A. Stochastic settings of the problems of Euclidean combinatorial optimization // Theory of stochastic processes. Vol. 9 (25), no. 3-4, 2003. – P. 170–175.
119. Роскладка О.В. Про класи еквівалентності комбінаторних многогранників // В кн.: X міжнародна наук. конф. ім. ак. М. Кравчука: Мат-ли конф. – К., 2004. – С. 537.
120. Роскладка О.В. Задачі оптимізації на полікомбінаторних множинах: властивості та розв'язування: Автореф. дис.... канд. фіз.-мат. наук: 01.05.01 / Інститут кібернетики НАН України. – К., 2005.– 16 с.
121. Роскладка О.В. Задачі оптимізації на полікомбінаторних множинах: властивості та розв'язування: Дис.... канд. фіз.-мат. наук: 01.05.01. – Полтава: ПНТУ, 2004. – 163 с.

122. Барболіна Т.М. Методи і алгоритми розв'язування оптимізаційних задач на розміщеннях з додатковими умовами: Автореф. дис... канд. фіз.-мат. наук: 01.05.01 / Інститут кібернетики НАН України. – К., 2005.– 19 с.
123. Барболіна Т.М. Методи і алгоритми розв'язування оптимізаційних задач на розміщеннях з додатковими умовами: Дис... канд. фіз.-мат. наук: 01.05.01. – Полтава: ПДПУ, 2005. – 141 с.
124. Новожилова М.В. Математичні моделі і методи розв'язання нелінійних задач розміщення геометричних об'єктів: Автореф. дис... д-ра фіз.-мат. наук: 01.05.02 / Інститут проблем машинобудування. НАН України. – Харків, 1999. – 35 с.
125. Шило В.П. Методи розв'язання складних задач дискретної оптимізації: Автореф. дис. ... д-ра фіз.-мат. наук: *01.05.01* / Інститут кібернетики ім. В.М. Глушкова НАН України. – К., 2003. – 33 с.
126. Бурдюк В.Я., Семенов В.А. Разрешимые случаи новой комбинаторной задачи оптимизации // Кибернетика и системный анализ. – 1999. – № 2. – С. 175–178.
127. Бурдюк В.Я., Сергеев С.О. О разбиении метрического пространства на непрерывные компоненты // Кибернетика и системный анализ. – 1999. – № 5. – С. 185–190.
128. Голуб И.В., Гребенник И.В., Кузьменко В.М. Комбинаторный подход к построению технологических штриховых кодов минимальной длины // Радиоэлектроника и інформатика. – 1998. – № 3. – С. 66–71.
129. Горунович С.А. Полиэдральная структура некоторых задач комбинаторной оптимизации. Автореф. дис. ... канд. физ.-мат. наук. Ин-т математики АН БССР. – Минск, 1982. – 16 с.
130. Айгнер М. Комбинаторная теория / Пер. с англ. – М.: Мир, 1982. – 402 с.
131. Баранов В.И., Стечкин Б.С. Экстремальные комбинаторные задачи и их приложения. – М.: Наука, 1989. – 160 с.
132. Корбут А.А., Финкельштейн Ю.Ю. Дискретное программирование. – М.: Наука, 1969. – 368 с.
133. Корбут А.А., Сигал И.Х., Финкельштейн Ю.Ю. Метод ветвей и границ: обзор теории, алгоритмов, программ и приложений. – Math. Operationsch und Statist., Ser. Optimiz. – 1977. – № 2. – P. 253–280.
134. Меламед И.И., Сигал И.Х. Исследование параметров алгоритмов ветвей и границ решения симметричной задачи коммивояжера // Автоматика и телемеханика. – 1997. – № 10. – С. 186–192.

135. Лихтенштейн В.Е. Модели и методы дискретного программирования. – М.: Наука, 1971. – 240 с.
136. Беллман Р. Динамическое программирование. – М.: Изд-во иностр. лит-ры, 1960. – 400 с.
137. Беллман Р., Дрейфус С. Прикладные задачи динамического программирования / Перевод с англ. / Под ред. Первозванского А.А. – М.: Наука, 1965. – 460 с.
138. Арис Р. Дискретное динамическое программирование. Введение в оптимизацию многошаговых процессов / Перевод с англ. Ю.П. Плетникова / Под ред. Поляка Б.Т. – М.: Мир, 1969. – 171 с.
139. Гроппен В.О. Оптимизация методов динамического программирования при решении экстремальных комбинаторных задач // Автоматика и телемеханика. – 1985. – № 12. – С. 79–84.
140. Held M., Karp R. M. A dynamic programming approach to sequencing problems // J. Soc. Industr. And Appl. Math. – 1962. – V. 10, № 1. – P. 196–210.
141. Cook W., Cunningham W., Pulleyblank W., Schrijver A. Combinatorial Optimization. John Wiley & Sons, 1997. – 368 p.
142. Babai L., Gal A., Kimmel P., Lokam S. Communication Complexity of Simultaneous Messages // SIAM Journal on Computing, 33(1), February 2004. – P. 137–166.
143. Butz M., Goldberg D., Stolzmann W. Probability-Enhanced Predictions in the Anticipatory Classifier System // Lecture Notes in Computer Science, Vol. 1996. – P. 37–48.
144. Reed P., Minsker B., Goldberg D. Simplifying multiobjective optimization: An automated design methodology for the nondominated sorted genetic algorithm-II // Water Resources Research, 39(7), July 2003. – P. 21–25.
145. Куценко І.А. Оцінки в задачах практичної стійкості дискретних систем та їх оптимізація: Автореф. дис. ... канд. фіз.-мат. наук: 01.05.02 / Київський національний ун-т ім. Тараса Шевченка. – К., 1995. – 22 с.
146. Khan N., Goldberg D., Pelikan M. Multi-Objective Bayesian Optimization Algorithm // Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2002), Morgan Kaufmann Publishers, July 2002. – P. 684.
147. Schrijver A.A Combinatorial Algorithm Minimizing Submodular Functions in Strongly Polynomial Time // JCTB: Journal of Combinatorial Theory, Series B, Vol. 80, 2000. – P. 346–355.
148. Зіньковська Ю.С. Дискретні екстремальні задачі в умовах невизначеності: питання стійкості. Автореф. дис. ... канд. фіз.-мат. наук. – Запоріжжя, 1997. – 16 с.

149. Miller A., Nemhauser G., Savelsbergh M. On the polyhedral structure of a multi-item production planning model with setup times // Mathematical Programming, Vol. 94, Number 2-3, 2002. – P. 375–405.
150. Тимофеева Н.К. Упорядочение множества значений аргумента целевой функции в комбинаторной оптимизации// Кибернетика и систем. анализ. – 1998. – № 6. – С. 78–87.
151. Cooper C., Gilchrist R., Kovalenko I.N., Novakovic D. Deriving the number of "good" permutations, with applications to cryptography // Кибернетика и системный анализ. – 1999. – № 5. – С. 10–17.
152. Farias R., Johnson E., Nemhauser G. A generalized assignment problem with special ordered sets: a polyhedral approach // Mathematical Programming, 89(1), 2000. – P. 187–203.
153. Farias R., Nemhauser G. A Polyhedral Study of the Cardinality Cosntrained Knapsack Problem // Mathematical Programming, 96(3), 2003. – P. 439–467.
154. Dantzig G. B. Discrete-variable extremum problems // Oper. Res. – 1957. – 5, № 2. – P. 266–277.
155. Gomory R. E. Outline of an algorithm for integer solution to linear programs. Bull. Amer. Math. Soc, 1958. – 64, № 5. – P. 275–278.
156. Gomory R. E. An algorithm for integer solutions to linear programs. Recent Advances Math. Program, McGraw-Hill Book Co., Inc., 1963. – P. 269–302.
157. Dalton R.E., Llewellyn R.W. An extension of the Gomory mixed-integer algorithm to mixed-discrebe variables, Manag. Sci., 1966. – 12. № 7. – P. 562–575.
158. *Финкельштейн* Ю.Ю. Алгоритм для решения задач целочисленного линейного программирования с булевыми переменными. // Экономика и мат. методы. – 1965. – № 5. – С. 746–759.
159. Реклейтис Г., Рейвиндран А., Рєгедел К. Оптимизация в технике. – М.: Мир, 1986. – Т. 1. – 352 с.
160. Land A.H., Doig A.G. An automatic method of solving discrete programming problems. Econometrica. – 1960.– 28, № 3. – P. 497–520.
161. Balas E. An additive algorithm for solving linear programs with zero-one variables. Operat. Res. – 1965. – 13, № 4. – P. 513–546.
162. Гудман С., Хидетниеми С. Введение в разработку и анализ алгоритмов. – М.: Мир, 1981. – 368 с.
163. Klee V.H. Walkup D.W. The d-step conjecture for polyhedra of dimension d<6 // Acta Math. – 1967. – V 117. № 1–2. – P. 323–327.
164. Chand D. R., Kapur S. S. An algorithm for convex polytopes // J. Assoc. Machinery. 1970. – V. 17. № 1. – P. 78–86.
165. Бренстед А. Введение в теорию выпуклых многогранников / Пер. с англ. – М.: Мир, 1988. – 240 с.

166. Черных О.Л. Построение выпуклой оболочки конечного множества точек на основе триангуляции // Журн. вычисл. математики и мат. физики. – 1991. – № 8. – С. 1231–1242.
167. Черных О.Л. Построение выпуклой оболочки конечного множества точек при приближенных вычислениях // Журн. вычисл. математики и мат. физики. – 1988. – № 9. – С. 1386–1396.
168. Voronoj G. Recherches sur le paralleloedres primitifs. – J. reine und angew. Math., 1909. – P. 61–70.
169. Voronoj G. Sur quelgues propriétes des formes quadratiques *positives parfaitess*. – *J. reine und angew*. Math., 1908. – P. 37–44.
170. Minkowski H. Theorie der konvexen Korper, Gesamm., Leipzig, 1911. – 308 p.
171. Вейль Г. Элементарная теория выпуклых многогранников. – В кн: Матричные игры. – М.: Физматгиз. – 1961. – 417 с.
172. Черников С.Н. Линейные неравенства. – М.: Наука, 1968. – 488 с.
173. Астафьев Н.Н. Линейные неравенства и выпуклость. – М.: Наука, 1982. – 216 с.
174. Чарин В.С. *Линейные преобразования и выпуклые множества*. – К.: Вища школа, 1978. – 366 с.
175. Русін В.В. Формування мережі і типів лікувально-профілактичних закладів в сучасних соціальних умовах великого міста (на прикладі м.Полтави): Автореф. дис. ... канд. Архітектури: 18.00.02 / Київський національний ун-т будівництва і архітектури. – К., 2000. – 18 с.
176. http://www2.msstate.edu/~dgh2/cvxpt.htm

ДОДАТКИ

Додаток А

Текст модуля програми мовою Object Pascal реалізації методів гілок і меж та динамічного програмування для евклідової комбінаторної задачі розміщення об'єктів обслуговування

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, ExtCtrls;

type
  TForm1 = class(TForm)
  Button1: TButton;
  Button3: TButton;
  OpenDialog1: TOpenDialog;
  Button5: TButton;
  Label1: TLabel;
  GroupBox1: TGroupBox;
  Button4: TButton;
  Button2: TButton;
  procedure Button3Click(Sender: TObject);
  procedure Button1Click(Sender: TObject);
  procedure Button4Click(Sender: TObject);
  procedure Button5Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);

  private
    {Private declarations}
  public
    {Public declarations}
  end;
const c=20;c1=500000;c2=500000;
//c1-константа, що задає максимальну кількість матриць X
//c2-константа, що задає максимальну кількість векторів Y

type
  vec2=array[1..c] of word;
  vec3=array[1..c2,1..c] of word;
```

```
  vec4=array[1..c2] of word;
  vec5=array[0..c,1..c2] of word;
  vec6=array[1..c,1..c] of word;
  vec7=array[0..c,1..c2] of longword;
  vec2Gb=array[1..c1,1..c,1..c] of word;
var
  Form1: TForm1; Time0: TDateTime;
  q1,a,kol:vec2;
  XDP,XDPnext:vec3;
  sumY:vec4;
  q,est:vec5;
  x0,r,x1:vec6;
  x:vec2Gb;
  y:vec3;
  QX:vec7;
i1,i2,i3,i4,i5,i6,i7,i8,i9,i10,i11,i12,i13,i14,i15,i16,i17,i18,i19, i20:word;
  quanX,quanXj,quanXjOpt:array[1..c] of longword;
  y1,Num:array[1..c2] of longword;
  Xj:array[0..c2,1..c] of word;
  MaxQuantityX,quanY:longword;
  s,t4,t10,n,rad,w,t1,t2,diapazon1,
  diapazon2,diapazon3,diapazon4:word;
  st:string[80];
  f,inf,test: textfile;

implementation

uses Unit2;
{$R *.dfm}
procedure TForm1.Button3Click(Sender: TObject);
begin
Halt;
end;

procedure PrintCondition;
var k,j:byte;
begin
Assignfile(f,'d:\Objects\Answers\answer.pas');
rewrite(f);
{Вивід даних у файл}
writeln(f,'*************** Початкові дані ******************');
writeln(f,'Кількість населених пунктів');writeln(f,n);
```

```
writeln(f,'Кількість клієнтів в кожному з населених пунктів');
for k:=1 to n do write(f,' ',a[k]);writeln(f);
writeln(f,'Матриця відстаней між населеними пунктами');
for k:=1 to n do begin for j:=1 to n do write(f,' ',r[k,j]); writeln(f) end;
writeln(f,'Радіус обслуговування'); writeln(f,rad);
writeln(f,'Кількість різних ємностей об"єктів обслуговування');
writeln(f,w);
writeln(f,'Величини об"єктів обслуговування');
for k:=1 to w do write(f,' ',q1[k]); writeln(f);
writeln(f);
{Введення початкових результатів закінчено}
closefile(f);
end;

procedure TForm1.Button1Click(Sender: TObject);
var inf:textfile;k,j:byte;
begin
if OpenDialog1.Execute and
FileExists(OpenDialog1.FileName) then
begin
Assignfile(inf,OpenDialog1.FileName);
Reset(inf);
end;
readln(inf,st);
{Кількість населених пунктів}
readln(inf,n);readln(inf,st);
s:=0;
{Кількість клієнтів в кожному з населених пунктів}
for k:=1 to n do begin read(inf,a[k]); s:=s+a[k]; end;
readln(inf);readln(inf,st);
{Матриця відстаней між населеними пунктами}
for k:=1 to n do begin for j:=1 to n do read(inf,r[k,j]); readln(inf) end;
readln(inf,st);
{Радіус обслуговування}
readln(inf,rad); readln(inf,st);
{Кількість різних ємностей об"єктів обслуговування}
readln(inf,w);readln(inf,st);
{Ємності об"єктів обслуговування}
for k:=1 to w do read(inf,q1[k]);
closefile(inf);
PrintCondition;
end;
```

```
{Процедура виводу на друк матриці}
procedure vyvid(p1:vec6);
var i,j:byte;
begin
append(f);
for i:=1 to n do begin
for j:=1 to n do
write(f,' ',p1[i,j]);
writeln(f);
        end;
closefile(f);
end;

procedure TForm1.Button5Click(Sender: TObject);
var k,j:byte;
begin
Assignfile(inf,'d:\temp\objects\Conditions\Cond_rand.pas');
reset(inf);
{Введення початкових даних, які генеруються випадковим чином
процедурою
randomize і оператором random}
readln(inf,st);readln(inf,n);
readln(inf,st);readln(inf,w);
readln(inf,st);readln(inf,diapazon1);
readln(inf,st);readln(inf,diapazon2);
readln(inf,st);readln(inf,diapazon3);
readln(inf,st);readln(inf,diapazon4);
randomize;
s:=0;
{Кількість клієнтів в кожному населеному пункті}
for k:=1 to n do begin a[k]:=random(diapazon1);s:=s+a[k]; end;
{Матриця відстаней між населеними пунктами}
for k:=1 to n do for j:=k+1 to n do r[k,j]:=random(diapazon2);
{Введеня діагональних та симетричних елементів}
for k:=1 to n do begin r[k,k]:=0; for j:=1 to k-1 do r[k,j]:=r[j,k] end;
{Радіус обслуговування}
rad:=random(diapazon3);
{Ємності об"єктів обслуговування}
for k:=1 to w do q1[k]:=random(diapazon4);
CloseFile(inf);
PrintCondition;
end;
```

```
{Блок процедур генерарації елементів полірозміщень Y}
procedure NewY;
var j:byte;
p:array[1..10] of byte;
begin
quanY:=quanY+1;
SumY[quanY]:=0;
p[1]:=i1;p[2]:=i2;p[3]:=i3;p[4]:=i4;p[5]:=i5;
p[6]:=i6;p[7]:=i7;p[8]:=i8;p[9]:=i9;p[10]:=i10;
for j:=1 to w do begin
Y[quanY,j]:=q[j,p[j]];sumY[quanY]:=sumY[quanY]+y[quanY,j];end;
end;

procedure combinationY;
var i,j:word;t:longword;
k3:word;
begin
append(f);
{s-загальна кількість клієнтів в усіх населених пунктах}
for i:=1 to w do begin
k3:=1;
   while (k3-1)*q1[i]<=s do begin
   q[i,k3]:=(k3-1)*q1[i];
      k3:=k3+1;
   end;
kol[i]:=k3-1;
write(f,'Q',i,'=[');
for j:=1 to kol[i] do begin
write (f,' ',q[i,j]);
                  end;
write(f,']');writeln(f,' kol[i]=',kol[i]);          end;
for i1:=1 to kol[1] do
for i2:=1 to kol[2] do
if w=2 then NewY else
for i3:=1 to kol[3] do
if w=3 then NewY else
for i4:=1 to kol[4] do
if w=4 then NewY else
for i5:=1 to kol[5] do
if w=5 then NewY else
for i6:=1 to kol[6] do
if w=6 then NewY else
```

```
for i7:=1 to kol[7] do
if w=7 then NewY else
for i8:=1 to kol[8] do
if w=8 then NewY else
for i9:=1 to kol[9] do
if w=9 then NewY else
for i10:=1 to kol[10] do
if w=10 then NewY;
writeln(f,'Кількість допустимих векторів Yj дорівнює ',quanY);
if quanY>c2 then begin
writeln(f,'Розмірність масиву допустимих векторів Yj перевищує 2 GB');halt;
          end;
closefile(f);
end;
{Кінець процедури combinationY}

{Процедура вибору оптимального вектора Y для рядка X}
procedure SelectYforX(vectorX:vec2;var m:word;var k:longword);
var j:byte;t:longword;
SumX:longword;
begin
SumX:=0;k:=0;m:=s;
{k-кількість оптимальних векторів Y для рядка vectorX, m.то кількість
оптимальних варіантів розміщення відділень в даному населеному пункті}
for j:=1 to n do
SumX:=SumX+vectorX[j];
for t:=1 to quanY do
if (SumX>=SumY[t])and(SumX-SumY[t]<m) then begin m:=SumX-SumY[t];
{Запам'ятовуємо номер t допустимого варіанту Y для рядка vectorX}
k:=t;           end

end;{procedure}

{Процедура формування матриці X0[nxn] макимально допустимих розташувань
об"єктів обслуговування.}
procedure GenerationX0;
var k,j:byte;no:vec2;
begin
```

```
append(f);
for k:=1 to n do
for j:=1 to n do
{Якщо пункт j "не дотягує" до пункту i, то на місці x0[i,j] повинен
стояти 0 }
if r[k,j]<=rad then x0[k,j]:=a[j] else x0[k,j]:=0;

writeln(f);
writeln(f,'Матриця максимально допустимих розташувань об"єктів
обслуговування');
writeln(f,'по населених пунктах');
vyvid(x0);
MaxQuantityX:=1;
for k:=1 to n do begin
No[k]:=0;
for j:=1 to n do if x0[k,j]<>0 then No[k]:=No[k]+1;
MaxQuantityX:=MaxQuantityX*No[k];
                end;
append(f);
writeln(f,'Максимальна кількість матриць X становить ',MaxQuantityX);
if MaxQuantityX>c1 then begin
writeln(f,'Розмірність масиву допустимих розв"язків перевищує 2 GB');
closefile(f);halt;
                end;
                closefile(f);
end;{procedure}

{Блок процедур генерарації сполучень з елементів матриці x0 - матриці
максимально допустимих розташувань об"єктів обслуговування.
Процедури
розраховані на відому наперед максимальну кількість об'єктів
обслуговування.
В данму випадку nmax=20. Результат: X-масив різних допустимих
матриць}

procedure PrintMatr(pr:vec6);
var i,j:byte;
begin
quanX[1]:=quanX[1]+1;
append(test);
if quanX[1] mod 1000 =0 then
```

```
writeln(test,'Формується матриця #',quanX[1],' із ',MaxQuantityX,' допустимих матриць');
closefile(test);
for i:=1 to n do for j:=1 to n do x[quanX[1],i,j]:=x1[i,j];
end;{procedure}

procedure matr(u,t:integer);
var k:byte;
begin
{Замінюємо на 0 всі елементи стовпця, крім розглядуваного}
for k:=1 to n do if k=u then X1[k,t]:=a[t] else X1[k,t]:=0;
end;

procedure GenerationX;
label 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20;
begin
append(f);
{Ініціалізуємо матрицю X1, яка буде містити допустимі розташування}
for i1:=1 to n do for i2:=1 to n do x1[i1,i2]:=0;

for i1:=1 to n do begin {1}
 if x0[i1,1]=a[1] then matr(i1,1) else goto 1;
for i2:=1 to n do begin {2}
 if x0[i2,2]=a[2] then matr(i2,2) else goto 2;
if n=2 then PrintMatr(X1) else
 for i3:=1 to n do begin {3}
 if x0[i3,3]=a[3] then matr(i3,3) else goto 3;
if n=3 then PrintMatr(X1) else
 for i4:=1 to n do begin {4}
 if x0[i4,4]=a[4] then matr(i4,4) else goto 4;
if n=4 then PrintMatr(X1) else
 for i5:=1 to n do begin {5}
 if x0[i5,5]=a[5] then matr(i5,5) else goto 5;
if n=5 then PrintMatr(X1) else
 for i6:=1 to n do begin {6}
 if x0[i6,6]=a[6] then matr(i6,6) else goto 6;
if n=6 then PrintMatr(X1) else
 for i7:=1 to n do begin {7}
 if x0[i7,7]=a[7] then matr(i7,7) else goto 7;
if n=7 then PrintMatr(X1) else
 for i8:=1 to n do begin {8}
 if x0[i8,8]=a[8] then matr(i8,8) else goto 8;
```

```
if n=8 then PrintMatr(X1) else
 for i9:=1 to n do begin {9}
  if x0[i9,9]=a[9] then matr(i9,9) else goto 9;
if n=9 then PrintMatr(X1) else
 for i10:=1 to n do begin {10}
  if x0[i10,10]=a[10] then matr(i10,10) else goto 10;
if n=10 then PrintMatr(X1) else
for i11:=1 to n do begin {11}
  if x0[i11,11]=a[11] then matr(i11,11) else goto 11;
if n=11 then PrintMatr(X1) else
 for i12:=1 to n do begin {12}
  if x0[i12,12]=a[12] then matr(i12,12) else goto 12;
if n=12 then PrintMatr(X1) else
 for i13:=1 to n do begin {13}
  if x0[i13,13]=a[13] then matr(i13,13) else goto 13;
if n=13 then PrintMatr(X1) else
 for i14:=1 to n do begin {14}
  if x0[i14,14]=a[14] then matr(i14,14) else goto 14;
if n=14 then PrintMatr(X1) else
 for i15:=1 to n do begin {15}
  if x0[i15,15]=a[15] then matr(i15,15) else goto 15;
if n=15 then PrintMatr(X1) else
 for i16:=1 to n do begin {16}
  if x0[i16,16]=a[16] then matr(i16,16) else goto 16;
if n=16 then PrintMatr(X1) else
 for i17:=1 to n do begin {17}
  if x0[i17,17]=a[17] then matr(i17,17) else goto 17;
if n=17 then PrintMatr(X1) else
 for i18:=1 to n do begin {18}
  if x0[i18,18]=a[18] then matr(i18,18) else goto 18;
if n=18 then PrintMatr(X1) else
 for i19:=1 to n do begin {19}
  if x0[i19,19]=a[19] then matr(i19,19) else goto 19;
if n=19 then PrintMatr(X1) else
 for i20:=1 to n do begin {20}
  if x0[i20,20]=a[20] then matr(i20,20) else goto 20;
if n=20 then PrintMatr(X1);
          {Початок формування нового вектора X}
          20:end;
          19:end;
          18:end;
          17:end;
```

```
        16:end;
        15:end;
        14:end;
        13:end;
        12:end;
        11:end;
        10:end;
        9:end;
        8:end;
        7:end;
        6:end;
        5:end;
        4:end;
        3:end;
        2:end;
        1:end;
writeln(f,'кількість допустимих матриць X=',quanX[1]);
closefile(f);
end;{procedure}
procedure OptimalMGM;
var m,l:word;x2,min:vec2;t,k:longword;i,j:byte;
begin
{Номер матриці перед початком галуження на 1-му рівні збігається
із початковим
номером матриці}
for t:=1 to quanX[1] do begin QX[1,t]:=t;est[0,QX[1,t]]:=0; end;

for i:=1 to n do begin{i}
append(f);
writeln(f,'********МАТРИЦІ РІВНЯ ',i,' ****************');
append(test);
writeln(test,'Обробляються матриці ',i,'-го рівня в кількості
',quanX[i],' штук');
closefile(test);
min[i]:=s;//min[i] - поточне значення мінімальної оцінки на рівні i
for t:=1 to quanX[i] do begin{t}
if t mod 1000=0 then
begin append(test);writeln(test,'Обробляється матриця
#',t);closefile(test);end;
{quanX[i] – кількість матриць X на рівні i}
for j:=1 to n do
```

```
x2[j]:=X[QX[i,t],i,j];
SelectYforX(x2,m,k);
est[i,QX[i,t]]:=est[i-1,QX[i,t]]+m;
//Початок відбору матриць для нового (i+1)-го рівня
if est[i,QX[i,t]]<min[i] then begin quanX[i+1]:=1;
min[i]:=est[i,QX[i,t]];
{Запам'ятовуємо номер t матриці X} QX[i+1,quanX[i+1]]:=QX[i,t];end
else
if est[i,QX[i,t]]=min[i] then begin quanX[i+1]:=quanX[i+1]+1;
QX[i+1,quanX[i+1]]:=QX[i,t];end;
          end;{t}
writeln(f,'quanX[',i,']=',quanX[i]);
writeln(f,'min(',i,')=',min[i]);
          end;{i}
writeln(f,'**********************************************');
writeln(f,**РЕЗУЛЬТАТ РОБОТИ ПРОГРАМИ ЗА МЕТОДОМ ГІЛОК І
МЕЖ**');
writeln(f,'**********************************************');
writeln(f,'Кількість оптимальних розміщень:',quanX[n+1]);
writeln(f,'Кількість клієнтів в усіх населених пунктах:',s);
writeln(f,'Кількість обслугованих клієнтів:',s-min[n]);
writeln(f,'Кількість клієнтів, обслуговування яких передається на
вищий рівень:',min[n]);
append(f);
for t:=1 to quanX[n+1] do begin{t}
writeln(f,'Матриця #',QX[n+1,t]);
for l:=1 to n do begin
for j:=1 to n do
x2[j]:=X[QX[n+1,t],l,j];
SelectYforX(x2,m,k);
for j:=1 to n do write(f,' ',X[QX[n+1,t],l,j]);
write(f,'  Y(опт):'); for j:=1 to w do write(f,' ',Y[k,j]);
writeln(f);
          end;
            end;
closefile(f);
end;{procedure}

procedure GenerationXj;
var m:word;x2,min:vec2;t,k:longword;i,j:byte;p,l:word;b:array[1..c]of byte;
begin
quanXj[1]:=0;
```

```
for k:=1 to n do begin{k}
for i:=1 to k do b[i]:=i;
p:=k;
while p>=1 do begin{while}
for i:=1 to n do begin
l:=1;
for j:=1 to k do l:=l*X0[i,b[j]];
if l<>0 then break;
                    end;
if l<>0 then begin
quanXj[1]:=quanXj[1]+1;
for j:=1 to n do Xj[quanXj[1],j]:=0;
for j:=1 to k do Xj[quanXj[1],b[j]]:=a[b[j]];
                    end;
if k=n then break;
if b[k]=n then
p:=p-1 else p:=k;
if p>=1 then
for i:=k downto p do
b[i]:=b[p]+i-p+1;
                    end;{while}
                         end;{k}
end;{procedure}

procedure OptimalMDP;
var m,sx:word;t,k,h:longword;i,j,l:byte;x2,min:vec2;bool,add:boolean;
label result;
begin
append(f);
i:=1;
for t:=1 to quanXj[1] do begin
for j:=1 to n do
x2[j]:=Xj[t,j];
SelectYforX(x2,est[1,t],Y1[t]);
XDP[t,1]:=t;          end;
{масив XDP[t,i] характеризує, з яких саме векторів Xj складається матриця;
t-номер матриці, i-номер рядка, тобто XDP[t,i]=h означає, що в матриці t
рядок номер i являє допустимий вектор Xj з номером h}

while i<=n do begin {while}
```

```
//writeln(f,'*****Допустимі матриці ',i,'-го рівня**********');
//for t:=1 to quanXj[i] do begin
//for l:=1 to i do begin
//for j:=1 to n do write(f,' ',Xj[XDP[t,l],j]);writeln(f);end;
//writeln(f,'************************************');
//              end;
//Пошук оптимальних матриць i-го рівня
min[i]:={est[i,1]}s;
QuanXjOpt[i]:=0;
//for t:=1 to quanXj[i] do
//writeln(f,'est[',i,',',t,']=',est[i,t]);

for t:=1 to quanXj[i] do begin
    if est[i,t]<min[i] then begin
    quanXjOpt[i]:=1; min[i]:=est[i,t];
    Num[quanXjOpt[i]]:=t;
              end
else    if est[i,t]=min[i] then begin
    quanXjOpt[i]:=quanXjOpt[i]+1;
    Num[quanXjOpt[i]]:=t;
              end;
                  end;
writeln(f,'*************РІВЕНЬ ',i,'************************');
writeln(f,'Допустимих матриць: ',quanXj[i],' Оптимальних матриць: ',quanXjOpt[i]);
writeln(f);

//writeln(f,'*****Оптимальні матриці ',i,'-го рівня**********');
//for t:=1 to quanXjOpt[i] do begin
//for l:=1 to i do begin
//for j:=1 to n do write(f,' ',Xj[XDP[Num[t],l],j]);writeln(f);end;
//writeln(f,'************************************');
//                 end;
if i=n then break;

//Формування матриць наступного рівня
quanXj[i+1]:=0;
for t:=1 to quanXjOpt[i] do begin
add:=false;//add характеризує чи додається до матриці t i-го рівня хоча б
//один вектор 1-го рівня
   for h:=1 to quanXj[1] do begin
```

```
    bool:=true;//bool характеризує чи додається до матриці t i-го рівня
//вектор 1-го рівня з порядковим номером h
     for j:=1 to n do begin
     sx:=0;
     for l:=1 to i do sx:=sx+Xj[XDP[Num[t],l],j];
     if Sx+Xj[h,j]>a[j] then begin bool:=false;break end;
           end;
     if bool then begin
     add:=true;
     quanXj[i+1]:=quanXj[i+1]+1;
     for l:=1 to i do XDPnext[quanXj[i+1],l]:=XDP[Num[t],l];
     XDPnext[quanXj[i+1],i+1]:=h;est[i+1,quanXj[i+1]]:=est[i,Num[t]]
     +est[1,h];
           end;
                end;{h}
if not add then goto result;
                end;{t}
for t:=1 to quanXj[i+1] do
for l:=1 to i+1 do XDP[t,l]:=XDPnext[t,l];

i:=i+1;//Перехід на наступний рівень
          end;{while}
result:writeln(f,'*********************************************');
writeln(f,'*****РЕЗУЛЬТАТ РОБОТИ ПРОГРАМИ ЗА МЕТОДОМ
ДИНАМІЧНОГО ПРОГРАМУВАННЯ*****');
writeln(f,'**********************************************');
writeln(f,'Кількість оптимальних розміщень:',quanXjOpt[i]);
writeln(f,'Кількість клієнтів в усіх населених пунктах:',s);
writeln(f,'Кількість обслугованих клієнтів:',s-min[i]);
writeln(f,'Кількість клієнтів, обслуговування яких передається на
вищий рівень:',min[i]);
writeln(f,'Оптимальні розміщення:');
for t:=1 to {quanXjOpt[i]}5 do begin
for l:=1 to i do begin
for j:=1 to n do write(f,' ',Xj[XDP[Num[t],l],j]);
write(f,' Y(опт):');
for j:=1 to w do write(f,' ',Y[Y1[XDP[Num[t],l]],j]);
writeln(f);   end;
writeln(f,'***********************************');
          end;
closefile(f);
end;{procedure}
```

```
//Реалізація методу гілок і меж
procedure TForm1.Button4Click(Sender: TObject);
begin
AssignFile(test, 'D:\Test.pas');
rewrite(test);
Time0:=Time;
//Формуємо множину допустимих векторів Y
combinationY;
//Формуємо множину допустимих матриць X0
GenerationX0;
//Формуємо множину допустимих матриць X
GenerationX;
//Обчислюємо оптимальні розташування об'єктів на кожному з n
рівнів
OptimalMGM;
Form1.Label1.Caption:='Роботу програми завершено! Час
виконання:'+TimeToStr(Time-time0);
append(f);
writeln(f,'Час виконання:',TimeToStr(Time-time0));
closefile(f);
end;

//Реалізація методу динамічного програмування
procedure TForm1.Button2Click(Sender: TObject);
begin
AssignFile(test, 'D:\Test.pas');
rewrite(test);
Time0:=Time;
//Формуємо множину допустимих векторів Y
combinationY;
//Формуємо множину допустимих матриць X0
GenerationX0;
//Формування допустимих векторів Xj
GenerationXj;
OptimalMDP;
Form1.Label1.Caption:='Роботу програми завершено! Час
виконання:'+TimeToStr(Time-time0);
append(f);
writeln(f,'Час виконання:',TimeToStr(Time-time0));
closefile(f);
end;{procedure}
end.
```

Додаток Б

Приклад розв'язування задачі методом гілок і меж

*************** Початкові дані ****************

Кількість населених пунктів

20

Кількість клієнтів в кожному з населених пунктів

20 15 10 7 9 12 8 50 25 18 23 45 19 37 6 32 18 51 35 7

Матриця відстаней між населеними пунктами

0	29	10	15	13	33	48	38	29	45	9	38	26	9	6	25	23	41	14	27
29	0	8	9	14	8	35	39	6	6	9	48	14	43	45	42	32	39	47	11
10	8	0	17	10	34	1	4	15	6	5	41	44	46	4	18	17	22	24	3
15	9	17	0	43	49	35	12	15	14	26	21	18	6	1	47	18	5	44	26
13	14	10	43	0	22	22	8	11	14	47	22	25	34	40	43	2	35	48	39
33	8	34	49	22	0	29	25	18	3	15	12	32	32	11	33	15	13	34	42
48	35	1	35	22	29	0	35	34	46	24	17	13	36	3	28	30	35	27	46
38	39	4	12	8	25	35	0	1	42	43	5	9	40	5	31	9	10	6	14
29	6	15	15	11	18	34	1	0	17	14	15	13	9	42	13	15	19	0	14
45	6	6	14	14	3	46	42	17	0	48	3	20	2	38	27	48	44	0	7
9	9	5	26	47	15	24	43	14	48	0	29	12	37	11	46	2	36	24	22
38	48	41	21	22	12	17	5	15	3	29	0	27	23	4	27	18	33	36	20
26	14	44	18	25	32	13	9	13	20	12	27	0	37	33	5	9	2	31	47
39	43	46	6	34	32	36	40	9	2	37	23	37	0	10	34	8	9	34	5
6	45	4	1	40	11	3	5	42	38	11	4	33	10	0	4	24	39	33	5
25	42	18	47	43	33	28	31	13	27	46	27	5	34	4	0	13	13	32	49
23	32	17	18	2	15	30	9	15	48	2	18	9	8	24	13	0	12	16	12
41	39	22	5	35	13	35	10	19	44	36	33	2	9	39	13	12	0	29	44
14	47	24	44	48	34	27	6	0	0	24	36	31	34	33	32	16	29	0	10
27	11	3	26	39	42	46	14	14	7	22	20	47	5	5	49	12	44	10	0

Радіус обслуговування

2

Кількість різних величин об"єктів обслуговування

4

Ємності об"єктів обслуговування

11 17 29 42

**************** Розрахунки ****************

Допустимі ємності відділень 1 типу (q1=11)

Q1=[0 11 22 33 44 55 66 77 88 99 110 121 132 143 154 165 176 187 198 209 220 231 242 253 264 275 286 297 308 319 330 341 352 363 374 385 396 407 418 429 440]   Кількість k[1]=41

Допустимі ємності відділень 2 типу (q2=17)

Q2=[0 17 34 51 68 85 102 119 136 153 170 187 204 221 238 255 272 289 306 323 340 357 374 391 408 425 442]   Кількість k[2]=27

Допустимі ємності відділень 3 типу (q3=29)
Q3=[0 29 58 87 116 145 174 203 232 261 290 319 348 377 406 435]
Кількість k[3]=16
Допустимі ємності відділень 4 типу (q4=42)
Q4=[0 42 84 126 168 210 252 294 336 378 420] Кількість k[4]=11

Кількість допустимих векторів Yj дорівнює 194832
Максимальна кількість допустимих матриць X становить 82944
**********************РІВЕНЬ 1********************
Кільість матриць на рівні [1]=82944   Оцінка на рівні [1]=3
**********************РІВЕНЬ 2********************
Кільість матриць на рівні [2]=82944   Оцінка на рівні [2]=7
**********************РІВЕНЬ 3********************
Кільість матриць на рівні [3]=82944   Оцінка на рівні [3]=7
**********************РІВЕНЬ 4********************
Кільість матриць на рівні [4]=20736   Оцінка на рівні [4]=7
**********************РІВЕНЬ 5********************
Кільість матриць на рівні [5]=5184   Оцінка на рівні [5]=7
**********************РІВЕНЬ 6********************
Кільість матриць на рівні [6]=1728   Оцінка на рівні [6]=8
**********************РІВЕНЬ 7********************
Кільість матриць на рівні [7]=1728   Оцінка на рівні [7]=9
**********************РІВЕНЬ 8********************
Кільість матриць на рівні [8]=1728   Оцінка на рівні [8]=9
**********************РІВЕНЬ 9********************
Кільість матриць на рівні [9]=1440   Оцінка на рівні [9]=9
**********************РІВЕНЬ 10*******************
Кільість матриць на рівні [10]=960   Оцінка на рівні [10]=9
**********************РІВЕНЬ 11*******************
Кільість матриць на рівні [11]=544   Оцінка на рівні [11]=9
**********************РІВЕНЬ 12*******************
Кільість матриць на рівні [12]=136   Оцінка на рівні [12]=9
**********************РІВЕНЬ 13*******************
Кільість матриць на рівні [13]=136   Оцінка на рівні [13]=9
**********************РІВЕНЬ 14*******************
Кільість матриць на рівні [14]=102   Оцінка на рівні [14]=9
**********************РІВЕНЬ 15*******************
Кільість матриць на рівні [15]=60   Оцінка на рівні [15]=11
**********************РІВЕНЬ 16*******************
Кільість матриць на рівні [16]=60   Оцінка на рівні [16]=14
**********************РІВЕНЬ 17*******************
Кільість матриць на рівні [17]=60   Оцінка на рівні [17]=14

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				Оптимальні ємності відділень – вектори Yj			
Оптимальне розташування №1																							
20																					*17*		
	15																			*11*			
					12															*11*			
		10				8															*17*		
							50	25													*17*	*58*	
									18				37					35		*22*	*68*		
											45									*11*	*34*		
												19					51			*11*	*17*		*42*
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
												19					51						
																			7				
Оптимальне розташування №2																							
20																					*17*		
	15																			*11*			

**********************РІВЕНЬ 18******************
Кільсть матриць на рівні [18]=60   Оцінка на рівні [18]=14
**********************РІВЕНЬ 19******************
Кільсть матриць *на рівні [19]=40   Оцінка на рівні* [19]=14
**********************РІВЕНЬ 20******************
Кільсть матриць на рівні [20]=8   Оцінка на рівні [20]=21
****

РЕЗУЛЬТАТ РОБОТИ ПРОГРАМИ
ЗА МЕТОДОМ ГІЛОК І МЕЖ
****

Кількість оптимальних розташувань: 8
Кількість клієнтів в усіх населених пунктах: 447
Кількість обслугованих клієнтів: 426
Кількість клієнтів, обслуговування яких передається на вищий рівень: 21

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**			
					12															*11*			
		10				8															*17*		
						50	25														*17*	*58*	
									18				37					35		*22*	*68*		
											45									*11*	*34*		
			7											6						*11*			
															32						*29*		
				9						23						18				*33*	*17*		
												19					51			*11*	*17*		*42*
																			7				
Оптимальне розташування №3																							
20																					*17*		
	15																			*11*			
					12															*11*			

Продовження додатка Б

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**			
		10				8															*17*		
							50	25										35			*68*		*42*
									18				37							*55*			
											45									*11*	*34*		
												19					51			*11*	*17*		*42*
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
																			7				
Оптимальне розташування №4																							
20																					*17*		
	15																			*11*			
					12															*11*			
		10				8															*17*		
							50	25													*17*	*58*	
									18				37					35		*22*	*68*		

Продовження додатка Б

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**			
											45									*11*	*34*		
												19					51			*11*	*17*		*42*
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
																			7				
Оптимальне розташування №5																							
20																					*17*		
	15																			*11*			
					12															*11*			
		10				8															*17*		
							50	25										35			*68*		*42*
									18				37							*55*			
											45									*11*	*34*		

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**			
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
												19					51			*11*	*17*		*42*
																			7				
Оптимальне розташування №6																							
20																					*17*		
	15																			*11*			
					12															*11*			
		10				8															*17*		
							50	25													*17*	*58*	
									18				37					35		*22*	*68*		
											45									*11*	*34*		
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
												19					51			*11*	*17*		*42*

Продовження додатка Б

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**			
																			7				
Оптимальне розташування № 7																							
20																					*17*		
	15																			*11*			
					12															*11*			
		10				8															*17*		
							50	25										35			*68*		*42*
											45									*11*	*34*		
												19					51			*11*	*17*		*42*
									18				37							*55*			
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
																			7				

Продовження додатка Б

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**			
Оптимальне розташування № 8																							
20																					*17*		
	15																			*11*			
					12															*11*			
		10				8															*17*		
							50	25										35			*68*		*42*
											45									*11*	*34*		
									18				37							*55*			
			7											6						*11*			
															32							*29*	
				9						23						18				*33*	*17*		
												19					51			*11*	*17*		*42*
																			7				

Час роботи програми 11 хв. 50 сек.

Додаток В

Приклад розв'язування задачі методом динамічного програмування

******************* Початкові дані *******************

Кількість населених пунктів

20

Кількість клієнтів в кожному з населених пунктів

20 15 10 7 9 12 8 50 25 18 23 45 19 37 6 32 18 51 35 11

Матриця відстаней між населеними пунктами

0	52	23	38	36	56	71	61	52	68	32	61	49	32	29	48	46	64	37	50
52	0	31	32	37	31	58	62	29	29	32	71	37	66	28	65	55	62	70	34
23	31	0	40	33	57	26	27	38	29	28	64	67	69	27	41	40	45	47	26
38	32	40	0	66	72	58	35	38	37	49	44	41	29	24	70	41	28	67	49
36	37	33	66	0	45	45	31	34	37	70	45	48	57	63	66	26	58	24	62
56	31	57	72	45	0	52	48	41	26	18	35	55	55	34	56	38	36	57	65
71	58	26	58	45	52	0	58	57	69	47	40	36	59	16	21	53	58	50	39
61	62	27	35	31	48	58	0	24	65	66	28	32	63	28	54	32	33	29	37
52	29	38	38	34	41	57	24	0	40	37	38	36	32	65	36	38	42	33	37
68	29	29	37	37	26	69	65	40	0	71	26	43	25	61	50	71	67	43	30
32	32	28	49	70	18	47	66	37	71	0	52	35	60	34	69	25	59	47	45
61	71	64	44	45	35	40	28	38	26	52	0	50	46	27	50	41	56	59	43
49	37	67	41	48	55	36	32	36	43	35	50	0	60	56	28	32	25	54	70
62	66	69	29	57	55	59	63	32	25	60	46	60	0	33	57	31	32	57	28
29	28	27	24	63	34	16	28	65	61	34	27	56	33	0	27	47	42	56	28
48	65	41	70	66	56	21	54	36	50	69	50	28	57	27	0	36	36	55	72
46	55	40	41	26	38	53	32	38	71	25	41	32	31	47	36	0	35	39	35
64	62	45	28	58	36	58	33	42	67	59	56	25	32	42	36	35	0	52	67
37	70	47	67	24	57	50	29	33	43	47	59	54	57	56	55	39	52	0	33
50	34	26	49	62	65	39	37	37	30	45	43	70	28	28	72	35	67	33	0

Радіус обслуговування

25

Кількість різних величин об"єктів обслуговування

10

Ємності об"єктів обслуговування

8 16 43 192 249 268 301 336 360 372

**************** Розрахунки ****************

Допустимі ємності відділень 1 типу (q1=8)

Q1=[0 8 16 24 32 40 48 56 64 72 80 88 96 104 112 120 128 136 144 152 160 168 176 184 192 200 208 216 224 232 240 248 256 264 272 280 288 296 304 312 320 328 336 344 352 360 368 376 384 392 400 408 416 424 432 440 448] Кількість k[1]=57
Допустимі ємності відділень 2 типу (q2=16)
Q2=[0 16 32 48 64 80 96 112 128 144 160 176 192 208 224 240 256 272 288 304 320 336 352 368 384 400 416 432 448] Кількість k[2]=29
Допустимі ємності відділень 3 типу (q3=43)
Q3=[0 43 86 129 172 215 258 301 344 387 430] Кількість k[3]=11
Допустимі ємності відділень 4 типу (q4=192)
Q4=[0 192 384] Кількість k[4]=3
Допустимі ємності відділень 5 типу (q5=249)
Q5=[0 249] Кількість k[5]=2
Допустимі ємності відділень 6 типу (q6=268)
Q6=[0 268] Кількість k[6]=2
Допустимі ємності відділень 7 типу (q7=301)
Q7=[0 301] Кількість k[7]=2
Допустимі ємності відділень 8 типу (q8=336)
Q8=[0 336] Кількість k[8]=2
Допустимі ємності відділень 9 типу (q9=360)
Q9=[0 360] Кількість k[9]=2
Допустимі ємності відділень 10 типу (q10=372)
Q10=[0 372] Кількість k[10]=2
Кількість допустимих векторів Yj дорівнює 3491136

***********************РІВЕНЬ 1*********************
Допустимих матриць: 36 Оптимальних матриць: 5
***********************РІВЕНЬ 2*********************
Допустимих матриць: 157 Оптимальних матриць: 16
***********************РІВЕНЬ 3*********************
Допустимих матриць: 448 Оптимальних матриць: 30
***********************РІВЕНЬ 4*********************
Допустимих матриць: 756 Оптимальних матриць: 24
***********************РІВЕНЬ 5*********************
Допустимих матриць: 552 Оптимальних матриць: 72
***********************РІВЕНЬ 6*********************
Допустимих матриць: 1392 Оптимальних матриць: 96
***********************РІВЕНЬ 7*********************
Допустимих матриць: 1392 Оптимальних матриць: 288
***********************РІВЕНЬ 8*********************
Допустимих матриць: 3696 Оптимальних матриць: 576

***********************РІВЕНЬ 9******************
Допустимих матриць: 6432 Оптимальних матриць: 576
***********************РІВЕНЬ 10******************
Допустимих матриць: 5472 Оптимальних матриць: 1440
***********************РІВЕНЬ 11******************
Допустимих матриць: 12384 Оптимальних матриць: 2304
***********************РІВЕНЬ 12******************
Допустимих матриць: 17856 Оптимальних матриць: 1728
***********************РІВЕНЬ 13******************
Допустимих матриць: 12096 Оптимальних матриць: 3456
***********************РІВЕНЬ 14******************
Допустимих матриць: 20736 Оптимальних матриць: 3456
***********************РІВЕНЬ 15******************
Допустимих матриць: 17280 Оптимальних матриць: 6912
***********************РІВЕНЬ 16******************
Допустимих матриць: 20736 Оптимальних матриць: 6912
***********************РІВЕНЬ 17******************
Допустимих матриць: 6912 Оптимальних матриць: 6912

На останніх трьох рівнях забудова не відбувається.
****

РЕЗУЛЬТАТ РОБОТИ ПРОГРАМИ ЗА МЕТОДОМ ДИНАМІЧНОГО ПРОГРАМУВАННЯ

****

Кількість оптимальних розміщень: 6912
Кількість клієнтів в усіх населених пунктах: 451
Кількість обслугованих клієнтів: 409
Кількість клієнтів, обслуговування яких передається на вищий рівень: 42

Перші 5 оптимальних розміщень

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				Оптимальні ємності відділень – вектори Yj									
Оптимальне розташування №1																													
						8														8									
															32						32								
																	51			8		43							
							50	25													32	43							
				9																8									
									23							18				8	32								
		10																		8									
								18													16								
										45												43							
											19										16								
																		35			32								
																			11	8									
20																					16								
					12															8									
													37								32								
			7											6						8									
	15																			8									
Оптимальне розташування №2																													
						8														8									
															32						32								
																	51			8		43							
							50	25													32	43							

Продовження додатка В

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**									
				9																*8*									
										23						18				*8*	*32*								
		10																		*8*									
									18												*16*								
											45											*43*							
												19									*16*								
																		35			*32*								
																			11	*8*									
20																					*16*								
					12															*8*									
			7												6					*8*									
													37								*32*								
	15																			*8*									
Оптимальне розташування №3																													
						8														*8*									
															32						*32*								
																	51			*8*		*43*							
							50	25													*32*	*43*							

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				**Оптимальні ємності відділень – вектори Yj**									
				9																*8*									
										23						18				*8*	*32*								
		10																		*8*									
									18												*16*								
											45											*43*							
												19									*16*								
																		35			*32*								
																			11	*8*									
					12															*8*									
20																					*16*								
													37								*32*								
			7											6						*8*									
	15																			*8*									
Оптимальне розташування №4																													
						8														*8*									
															32						*32*								
																	51			*8*		*43*							
							50	25													*32*	*43*							
				9																*8*									
										23						18				*8*	*32*								
		10																		*8*									

<table>
<tr><th colspan="20">Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X</th><th colspan="10">Оптимальні ємності відділень – вектори Yj</th></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>18</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>16</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>45</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>43</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>19</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>16</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>35</td><td></td><td></td><td>32</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>11</td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td>12</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>20</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>16</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>6</td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>37</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>32</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>15</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="30">Оптимальне розташування №5</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>32</td><td></td><td></td><td></td><td></td><td></td><td>32</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>51</td><td></td><td></td><td>8</td><td></td><td>43</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>50</td><td>25</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>32</td><td>43</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td>9</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>23</td><td></td><td></td><td></td><td></td><td></td><td>18</td><td></td><td></td><td></td><td>8</td><td>32</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td>10</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>8</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>18</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>16</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>45</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>43</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

Продовження додатка В

Оптимальне розміщення клієнтів по населених пунктах (номер рядка – номер пункту) – матриці X																				Оптимальні ємності відділень – вектори Yj									
												19									*16*								
																			11	*8*									
																		35			*32*								
20																					*16*								
					12															*8*									
													37								*32*								
			7											6						*8*									
	15																			*8*									

Час роботи програми 50 сек.

Наукове видання

ЄМЕЦЬ Олег Олексійович
РОСКЛАДКА Олена Володимирівна

ЗАДАЧІ ОПТИМІЗАЦІЇ НА ПОЛІКОМБІНАТОРНИХ МНОЖИНАХ: властивості та розв'язування

Головний редактор ***Деркач Є.А.***
Редактор-коректор ***Іващенко Т.П.***
Комп'ютерна верстка ***Корніліч О.С.***

Здано до редакції 20.05.06 року. Підписано до друку 27.09.2006 року. Формат 148×210. Папір 60 г/м². Ум.друк.арк. 9,0 + 0,1 (обкл.). Тираж 1 000 прим. Зам. №1152

Видано редакційно-видавничим центром ПУСКУ
36014, м. Полтава, вул. Коваля, 3, к. 115, ☎ 8 (0532) 50-24-81
e-mail: rio@uccu.org.ua

Свідоцтво про внесення до Державного реєстру видавців, виготівників і розповсюджувачів видавничої продукції ДК № 2288 від 5.09.2005 р.

ЄМЕЦЬ ОЛЕГ ОЛЕКСІЙОВИЧ

Доктор фізико-математичних наук, професор, завідувач кафедри математичного моделювання та соціальної інформатики Полтавського університету споживчої кооперації України. Підготував 9 кандидатів наук за спеціальностями: теоретичні основи інформатики та кібернетики, математичне моделювання. Має понад 200 наукових публікацій, у тому числі 4 монографії та два навчальних посібника.

РОСКЛАДКА ОЛЕНА ВОЛОДИМИРІВНА

Кандидат фізико-математичних наук за спеціальністю теоретичні основи інформатики та кібернетики, доцент кафедри математичного моделювання та соціальної інформатики Полтавського університету споживчої кооперації України. Має близько 20 наукових праць та навчально-методичних розробок.