

**НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ  
ІНСТИТУТ ПРОБЛЕМ МАШИНОБУДУВАННЯ ім. А.М. ПІДГОРНОГО**

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ**

**І.В. ГРЕБЕННІК, Т.Є. РОМАНОВА, І.А. УРНЯЄВА,  
В.М. ДУБІНСЬКИЙ, Н.С. КАЛАЙДА**

**ОПТИМІЗАЦІЯ БАЛАНСНОГО І КЛАСТЕРНОГО  
КОМПОНУВАННЯ ТА ЗАВАНТАЖЕННЯ  
КОНТЕЙНЕРІВ**

Київ Наукова думка 2021

УДК 519.8

Монографію присвячено актуальній проблемі створення сучасних комп'ютерних технологій (математичних моделей, методів, алгоритмів та програм), орієнтованих на розв'язання задач оптимізації компонування та завантаження контейнерів з урахуванням технологічних обмежень. Досліджено три взаємопов'язані класи оптимізаційних задач: розрідженого компонування кластерів геометричних об'єктів у контейнері; балансного компонування 3D-об'єктів у контейнері, розділеному горизонтальними стелажми; планування завантаження контейнерів. Побудовано математичні моделі у вигляді задач математичного програмування. Враховано комбінаторні властивості зазначених задач, геометричні обмеження на допустимі відстані між об'єктами (кластерами об'єктів), зони заборони та умови балансу. Запропоновано та обґрунтовано стратегії розв'язання. Розглянуто практичні застосування, наведено результати обчислювальних експериментів.

Для науково-технічних співробітників, викладачів, аспірантів і студентів вищих навчальних закладів, що спеціалізуються у галузі математичного моделювання, дослідження операцій, оптимізації складних систем.

Рецензенти:

*О.М. Кисельова*, член-кореспондент НАН України,  
доктор фізико-математичних наук, професор,  
*П.І. Стецюк*, доктор фізико-математичних наук,  
старший науковий співробітник

*Рекомендовано до друку*  
*вченою радою Інституту проблем машинобудування ім. А.М. Підгорного НАН*  
*України*  
*(протокол № 9 від 9 вересня 2021 р.),*  
*вченою радою Харківського національного університету радіоелектроніки*  
*(протокол № 8 від 24 вересня 2021 р.)*

Науково-видавничий відділ фізико-математичної та технічної літератури  
Редактор *В.В. Вероцька*

**ISBN 978-966-00-1834-1**

© І.В.Гребеннік, Т.Є. Романова, І.А.Урняєва,  
В.М. Дубінський, Н.С. Калайда, 2021

## ЗМІСТ

<b>ВСТУП .....</b>	<b>4</b>
<b>РОЗДІЛ 1. ЗАДАЧІ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ.....</b>	<b>7</b>
1.1. СИСТЕМНІ АСПЕКТИ ПРОБЛЕМИ ЗАСТОСУВАННЯ КОНТЕЙНЕРІВ .....	7
1.2. ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ У ТРАНСПОРТНІЙ ЛОГІСТИЦІ.....	11
1.3. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТА РОЗВ'ЯЗАННЯ ЗАДАЧ ОПТИМАЛЬНОГО ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ .....	18
1.4. ПОСТАНОВКА ЗАДАЧІ БАЛАНСНОГО КОМПОНУВАННЯ (BLP).....	32
1.5. ОСНОВНІ ВИЗНАЧЕННЯ МЕТОДУ РНІ-ФУНКЦІЙ .....	38
1.6. ЗАГАЛЬНА ЗАДАЧА ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ .....	41
<b>РОЗДІЛ 2. ОПТИМІЗОВАНЕ ПАКУВАННЯ КЛАСТЕРІВ ОБ'ЄКТІВ         У ПРЯМОКУТНОМУ КОНТЕЙНЕРІ.....</b>	<b>43</b>
2.1. КЛАСТЕРИЗАЦІЯ ПАРИ НЕРЕГУЛЯРНИХ ОБ'ЄКТІВ .....	43
2.2. ЗАДАЧА КОМПОНУВАННЯ КЛАСТЕРІВ ОБ'ЄКТІВ .....	58
2.3. ЗАСОБИ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ КОМПОНУВАННЯ КЛАСТЕРІВ ОБ'ЄКТІВ У ПРЯМОКУТНОМУ КОНТЕЙНЕРІ .....	61
2.4. МАТЕМАТИЧНА МОДЕЛЬ ЗАДАЧІ КОМПОНУВАННЯ КЛАСТЕРІВ ОБ'ЄКТІВ .....	69
2.5. АЛГОРИТМ РОЗВ'ЯЗАННЯ ЗАДАЧІ КОМПОНУВАННЯ КЛАСТЕРІВ ОБ'ЄКТІВ .....	71
2.6. ПРИКЛАДИ КОМПОНУВАННЯ КЛАСТЕРІВ У ПРЯМОКУТНОМУ КОНТЕЙНЕРІ.....	74
2.7. ОПТИМІЗОВАНЕ ПАКУВАННЯ КЛАСТЕРІВ ЗА УМОВИ БАЛАНСУ .....	85
<b>РОЗДІЛ 3. ОПТИМІЗАЦІЙНІ ЗАДАЧІ БАЛАНСНОГО         КОМПОНУВАННЯ ВАНТАЖІВ У КОНТЕЙНЕРАХ ...</b>	<b>107</b>
3.1. КОМБІНАТОРНІ КОНФІГУРАЦІЇ В ОПТИМІЗАЦІЙНИХ ЗАДАЧАХ БАЛАНСНОГО КОМПОНУВАННЯ КОНТЕЙНЕРІВ .....	107
3.2. МАТЕМАТИЧНА МОДЕЛЬ КОМБІНАТОРНОЇ ЗАДАЧІ БАЛАНСНОГО КОМПОНУВАННЯ.....	114

3.3. МЕТОД РОЗВ'ЯЗАННЯ КОМБІНАТОРНОЇ ЗАДАЧІ БАЛАНСНОГО КОМПОНУВАННЯ.....	117
3.4. РЕЗУЛЬТАТИ ЧИСЛОВИХ ЕКСПЕРИМЕНТІВ .....	120
3.5. КОМБІНАТОРНА ЗАДАЧА БАЛАНСНОГО КОМПОНУВАННЯ ЗІ СПЕЦІАЛЬНИМИ ГЕОМЕТРИЧНИМИ ОБМЕЖЕННЯМИ ТА УМОВАМИ БАЛАНСУ .....	127
3.6. БАГАТОКРИТЕРІАЛЬНА ЗАДАЧА БАЛАНСНОГО КОМПОНУВАННЯ.....	137
<b>РОЗДІЛ 4. ЗАДАЧІ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ НА ЗАЛІЗНИЧНОМУ ТЕРМІНАЛІ .....</b>	<b>147</b>
4.1. КОНТЕЙНЕРНІ ОПЕРАЦІЇ НА ЗАЛІЗНИЧНОМУ ТЕРМІНАЛІ .....	147
4.2. ЗАДАЧА СКЛАДАННЯ РОЗКЛАДУ РУХУ ВАНТАЖНИХ ПОТЯГІВ ТА ОБРОБЛЕННЯ ВАНТАЖІВ НА ЗАЛІЗНИЧНОМУ ТЕРМІНАЛІ.....	150
4.3. ЗАДАЧА РОЗМІЩЕННЯ КОНТЕЙНЕРІВ ЗА УМОВ ЕКСПЛУАТАЦІЇ ЗАЛІЗНИЧНОГО ТЕРМІНАЛА .....	154
4.4. РОЗМІЩЕННЯ КОНТЕЙНЕРІВ РІЗНОГО ТИПУ НА ЗАЛІЗНИЧНОМУ КОНТЕЙНЕРНОМУ ТЕРМІНАЛІ ЗА ДОПОМОГОЮ DOUBLE-STACK ЗАВАНТАЖЕННЯ.....	160
4.5. ОПТИМІЗАЦІЯ ПОРЯДКУ ПЕРЕМІЩЕННЯ КОНТЕЙНЕРІВ НА ЗАЛІЗНИЧНИХ ТЕРМІНАЛАХ.....	169
4.6. ЗАСТОСУВАННЯ ЗАДАЧ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ У ЛОГІСТИЧНИХ ПРОЦЕСАХ ПОПЕРЕДЖЕННЯ ТА ЛІКВІДАЦІЇ НАДЗВИЧАЙНИХ СИТУАЦІЙ .....	174
<b>ВИСНОВКИ .....</b>	<b>178</b>
<b>СПИСОК УМОВНИХ СКОРОЧЕНЬ .....</b>	<b>180</b>
<b>СПИСОК ЛІТЕРАТУРИ.....</b>	<b>181</b>

## ВСТУП

За останні десятиліття контейнер як базова компонента концепції одиничного завантаження набув актуального значення для міжнародних морських та сухопутних вантажних перевезень. Завдяки постійному збільшенню масштабів контейнеризації кількість та потужність контейнерних терміналів у морських портах і на залізничних станціях та конкуренція між ними досягли високого рівня. Сьогодні контейнерні операції неможливі без ефективного використання інформаційних технологій, а також відповідних методів оптимізації та дослідження операцій.

У зв'язку з цим великої значущості набувають задачі пакування, що застосовуються в інженерії (серед них Container Loading, Vehicle Loading, Layout problem), які виникають у разі компонування та завантаження сухопутного, водного та повітряного транспорту: кораблів, підводних човнів, потягів, вантажних автомобілів, транспортних літаків, космічних ракет, космічних апаратів тощо. Ці задачі належать до класу задач тривимірного пакування (3D-Packing Problem). Їх можна охарактеризувати множиною обмежень розміщення та додатковими обмеженнями, серед яких пріоритет розміщуваних об'єктів, орієнтація об'єктів, обмеження ваги, рівноваги, стійкості тощо.

Один із напрямів сучасних досліджень у галузі пакування присвячений задачам балансного компонування (Balance Layout Problems, BLP). Ці задачі також називають задачами розміщення з додатковими обмеженнями (Constraint Layout Problems). Додатковими є обмеження поведінки (behavior constraints), які стосуються технічних характеристик системи (performance constraints).

Як наслідок, суттєвого теоретичного та прикладного значення набуває проблема розроблення інтелектуальних технологій розв'язання оптимізаційних задач розміщення, що базуються на конструктивних засобах математичного та комп'ютерного моделювання та сучасних методах

оптимізації варіантів розміщення 3D-об'єктів, які адаптуються для конкретних галузей застосування та технологічних процесів. Це дає змогу отримати оптимальний варіант розміщення згідно з заданими критеріями якості за наявними вихідними даними.

Розробленню інтелектуальних комп'ютерних технологій у галузі системного аналізу, дослідженню операцій, оптимізації, математичного моделювання, штучного інтелекту присвячені праці вітчизняних учених, зокрема, І.В. Сергієнка [1–3], М.З. Згуровського [4], С.І. Ляшка [5, 6], Л.Ф. Гуляницького [7], Н.В. Семенової [8–10], П.І. Стецюка [11–12], Ю.В. Крака [13], В.В. Грицика, А.І. Шевченка, О.М. Кісельової [14], В.П. Шила [1, 15], П.І. Бідюка [16], Р.П. Базилевича [17], Г. П. Донця [18] та багатьох інших.

Ефективність функціонування інтелектуальних систем розв'язання задач розміщення 3D об'єктів залежить від наявності в них можливостей автоматичної побудови математичних моделей. Ці можливості доволі складно реалізувати формально. Більшість сучасних комп'ютерних систем для розв'язання оптимізаційних задач розміщення орієнтуються на заздалегідь побудовані математичні моделі задач, які слід подати у специфічних форматах. Це вимагає створення нових інтелектуальних комп'ютерних технологій моделювання оптимізаційних задач розміщення в різних галузях науки, техніки та промисловості.

Ефективним засобом аналітичного моделювання відношень геометричних об'єктів в оптимізаційних задачах розміщення є метод  $\rho$ -функцій, запропонований Ю. Г. Стояном, який очолює українську наукову школу з геометричного проектування [19, 20].

У науковій школі Ю. Г. Стояна дослідження його учнів, серед яких М. І. Гіль, В. М. Комяк, М. В. Новожилова, Т. Є. Романова, О. В. Панкратов, В. М. Пацук, Г. М. Яськов, А. М. Чугай, присвячені створенню засобів математичного моделювання, математичних моделей і методів розв'язання задач розміщення об'єктів [20–29]. Комбінаторні оптимізаційні задачі геометричного проектування та методи їх розв'язання досліджено в працях

С. В. Яковлева [19, 20, 26, 30–34], О. О. Ємця [35–37], І. В. Гребенніка [38, 39] та їх учнів (див., зокрема, [40]).

Отже, актуальними є розроблення та розвиток сучасних комп'ютерних технологій (математичних моделей, методів, алгоритмів та програм) для розв'язання задач проектування оптимального розміщення вантажів заданих форм та розмірів у контейнерах з урахуванням геометричних обмежень та умов балансу.

Автори монографії ставили за мету проаналізувати сучасні оптимізаційні моделі, методи та алгоритми для підвищення ефективності розв'язання задач завантаження контейнерів та навести рекомендації щодо розроблення методів оптимізації балансного і кластерного компонування та завантаження контейнерів. Деякі питання, наведені в монографії, попередньо розглянуто в працях [41–58].

## РОЗДІЛ 1. ЗАДАЧІ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ

### 1.1. Системні аспекти проблеми застосування контейнерів

Контейнерні перевезення є основою матеріальних потоків у світовій економіці як на глобальному, так і на регіональних та місцевих рівнях.

Контейнеризацію можна визначити як складову технічного прогресу в процесі реалізації транспортних перевезень, складуванні та зберіганні різнорідних вантажів, що сприяє підвищенню ефективності логістичних процесів в економічних системах. Процес контейнеризації, що розпочався у середині 50-х років ХХ століття, зумовив кардинальні перетворення у світовій торгівлі. Притаманна контейнеризації модель доставки товарів «від дверей до дверей» сприяла багаторазовому збільшенню обсягів і номенклатури вантажів, які перевозяться в контейнерах, і підтриманню розвитку глобального ринку товарів.

У відомих літературних джерелах так охарактеризовано актуальність контейнерних перевезень:

«Завжди швидші та економічніші перевезення з використанням логістичних систем, що охоплюють всю земну кулю, дали змогу досягти відомого на сьогодні розподілу праці. Це синергетичні процеси, які і надалі сприятимуть більш ніж середнім темпам зростання контейнерних перевезень» [59].

«Контейнер становить основу високоавтоматизованої системи переміщення вантажів з будь-якого місця та в будь-яке місце з мінімальними витратами та ускладненнями на шляху. Контейнер зробив доставку дешевою, і цим змінив форму світової економіки» [60].

Питання достатньої доступності контейнерного обладнання постає в різних контекстах, загальний аналіз найважливіших факторів, які визначають достатність чи обмеженість поставок, є актуальним. Огляд Всесвітньої ради судноплавства (WSC) надає перелік факторів, що впливають на розмір та

використання глобального парку контейнерного обладнання, а також поточний стан сукупної пропозиції порівняно із попитом [61].

Контейнерні перевезення є важливою складовою системи глобальних та регіональних транспортних перевезень, яку інтегровано в усі світові економічні та геополітичні процеси. Ці процеси суттєво впливатимуть на розвиток контейнерних перевезень у найближчій та середньостроковій перспективі. У Підсумковому звіті Генерального директорату з регіональної та міської політики Європейської комісії за 2019 рік [62] зазначається:

1. На розвиток залізничного контейнерного транспорту впливають різні кластери факторів, такі як: глобальні мегатренди, макроекономічні тенденції, технологічний прогрес, глобальні та європейські транспортні тенденції, а також конкурентне регіональне середовище залізничного ринку та розвиток залізничної інфраструктури.

2. Геополітичні та економічні, а також технологічні та екологічні фактори матимуть найважливіший вплив на розвиток ринку контейнерних перевезень у наступні десять років як у глобальному, так і в національному масштабі.

Аналізуючи фактори зовнішнього середовища, що впливають у межах глобального ринку на процеси транспортування контейнерів, у [62] наведено карту взаємозв'язків світових ризиків та тенденцій 2019 (рис. 1.1), на якій показано роль і місце контейнерних перевезень у зазначених процесах.

Масштаби глобальних транспортних потоків, неможливих без ефективної системи контейнерних перевезень, зображено на діаграмі, наведеній на рис. 1.2 [62]. Прогнозується, що в 2019–2023 роках пропускна здатність глобальних контейнерних морських портів зростатиме на 5,5% щороку, досягаючи майже 1000 мільйонів TEU (Twenty-foot Equivalent Unit – умовна одиниця вимірювання місткості контейнерів).

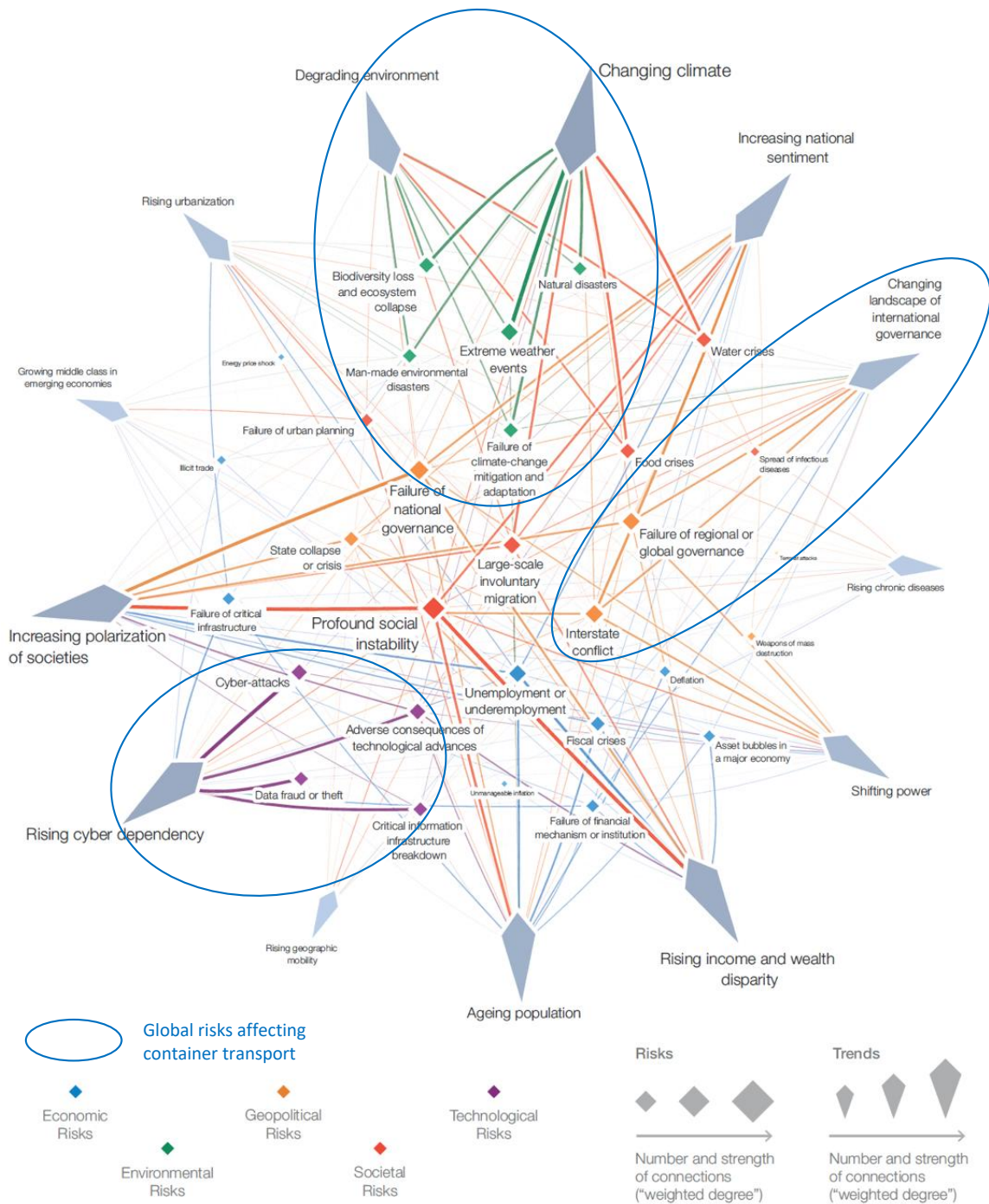


Рис. 1.1. Карта взаємозв'язків ризиків та тенденцій 2019<sup>1</sup> [62]

<sup>1</sup> Тут та далі деякі рисунки з англomовних джерел наведено зі збереженням мови оригіналу в текстових елементах з відповідними посиланнями.



**Рис. 1.2.** Основні світові потоки імпорту, 2017 р. (млрд дол. США) [62]

Згідно з обсягами глобальних контейнерних перевезень окремо слід розглядати проблему забезпечення ефективного завантаження одного контейнера. Адже ефект від оптимізації завантаження одного контейнера можна помножити на сотні мільйонів реальних контейнерів, задіяних у світових контейнерних перевезеннях. Сумарний ефект від оптимізації завантаження контейнерів полягає в зменшенні кількості контейнерів для перевезення того самого обсягу вантажів та, отже, у зниженні загальної вартості перевезень.

На підставі наведеного можна зробити такі висновки: контейнерні перевезення є підсистемою глобальної економічної та геополітичної системи, яка значно впливає на світові процеси; контейнерні перевезення мають тенденцію до кількісного зростання та технологічного розвитку на всіх рівнях – глобальному, регіональному та місцевому; оптимізацію завантаження контейнерів варто розглядати як на етапі виконання операцій переміщення контейнерів у контейнерному терміналі, так і на етапі завантаження окремих контейнерів, що є джерелом суттєвого економічного ефекту.

## 1.2. Завантаження контейнерів у транспортній логістиці

Наведемо основні поняття, необхідні для аналізу проблеми завантаження контейнерів на етапі виконання операцій переміщення контейнерів у контейнерному терміналі, відповідно до [63–67].

Інтермодальні контейнерні перевезення – це перевезення контейнерів від початкової точки до місця призначення щонайменше двома видами транспорту, перехід між якими здійснюється на інтермодальному контейнерному терміналі [63]. Прикладом інтермодальних контейнерних перевезень є транспортування контейнерів комбінацією вантажних, залізничних і морських транспортних засобів. Інтермодальні контейнерні перевезення належать до мультимодального ланцюга контейнерно-транспортних послуг. Цей ланцюг поєднує відправника з кінцевим одержувачем і простягається на великі відстані. Контейнерні перевезення є основним компонентом інтермодальних перевезень [63].

Ефективна організація інтермодальної системи суттєво впливає на якість транспортних послуг, тому використання інформаційних технологій є важливою умовою розвитку транспортних логістичних систем [68].

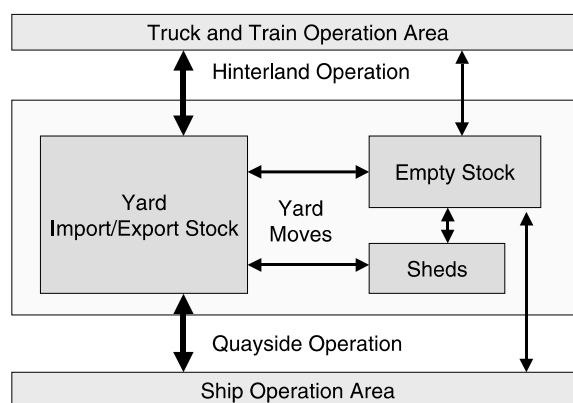
Основне завдання, яке вирішують інтермодальні контейнерні перевезення – це оптимізація часу та вартості переміщення вантажів за рахунок перевантаження контейнерів з одного виду транспорту на інший [68].

Контейнерні термінали – це об'єкти проміжного призначення, які дають змогу транспортним контейнерам змінювати способи транспортування на шляху до їх кінцевого місця призначення. Часто вантаж прибуває на контейнерний термінал на одному судні і розподіляється по декількох видах транспорту для доставки внутрішнім клієнтам. Термінал також є майданчиком, призначеним для обслуговування та тимчасового складування транспортних контейнерів. Іноді в терміналі здійснюються розвантаження, завантаження і зберігання вантажу всередині контейнерів [67].

Загалом, контейнерні термінали можна описати як відкриті системи для оброблення матеріальних потоків із двома зовнішніми інтерфейсами. Під

зовнішніми інтерфейсами розуміють причал для завантаження і розвантаження кораблів та узбережжя, де контейнери завантажуються і розвантажуються на/або з вантажних автомобілів та поїздів. Контейнери зберігаються в зонах зберігання у штабелях, що полегшує розділення прибережної та наземної експлуатації термінала.

Після прибуття в порт контейнерне судно відводиться на причал, обладнаний кранами для завантаження та вивантаження контейнерів. Нерозвантажені контейнери транспортуються в зони зберігання поблизу місця, де вони будуть перевантажені. Контейнери, що прибувають автотранспортом або залізницею до термінала, перевозять у межах районів експлуатації вантажних автомобілів та поїздів. Їх забирає внутрішнє обладнання термінала та розподіляє по відповідних ділянках зони зберігання. Якщо в терміналі є порожні склади, виконуються додаткові переміщення; ці кроки охоплюють перевезення між порожніми стапелями, пакувальним центром та імпортними й експортними контейнерними стапелями (рис. 1.3).



**Рис. 1.3.** Зони експлуатації контейнерного термінала морського порту та потік транспорту [65]

Зона зберігання контейнерів, зазвичай, розділена на різні блоки, які поділяються на ряди, бухти та яруси. Деякі ділянки штабелів зарезервовані для спеціальних контейнерів, таких як рефрижератори, що потребують

електричного підключення; небезпечних вантажів або контейнерів для перевантаження, для яких заборонено складати штабелі.

Автоматизація завантаження контейнерів використовується в європейській залізничній вантажній галузі. Великі термінали розглядають, планують і частково використовують технології автоматизації для таких процесів, як штабелювання і транспортування контейнерів. Повна автоматизація рейкових кранів є дуже складним завданням через вимоги техніки безпеки у разі взаємодії людини і машини. Через це повна автоматизація внутрішніх терміналів поки що не дуже поширена [69].

Контейнер є транспортним засобом або оборотною тарою, що забезпечує зручність транспортування і зберігання вантажу. Економічні фактори стимулюють накопичення контейнерів у великі партії для перевезення по морю спеціалізованими судами-контейнеровозами місткістю у тисячі контейнерів. Вартість таких судів визначає вимоги до ефективності вантажно-розвантажувальних операцій: максимально швидке обслуговування суден є основним завданням морського терміналу. Наземні транспортні засоби, на яких підвозяться контейнери в порт і вивозяться з порту, мають на порядки менші місткості: потяг – близько 100 контейнерів, вантажний автомобіль – 1–2 контейнери. Отже, обслуговуванням одного контейнерного судна займаються десятки потягів і сотні автомобілів [70]. Робота автомобільного та залізничного вантажного транспорту в морському терміналі має бути чітко організованою та синхронізованою. Так само слід розглядати й основні завдання залізничних та автомобільних контейнерних терміналів – мінімізація часу виконання операцій оброблення контейнерів з дотриманням відповідних технологічних вимог.

Контейнерні перевезення – вид вантажних перевезень, який застосовується практично на всіх видах транспорту. Контейнери, що найчастіше використовуються, є 20- (рис. 1.4) і 40-футовими з їх різновидами, які відповідають вимогам ISO [71].



**Рис. 1.4.** Стандартний 20-футовий контейнер [71]

Контейнери стандартизовані із зазначенням вимірювання внутрішнього простору в квадратних футах і двох латинських літер, що позначають їх тип. Для визначення обсягів контейнерних перевезень використовують два способи вимірювання: TEU і FEU. Перший позначає Twenty-foot Equivalent Unit, тобто кількість контейнерів в одиницях вимірювання, рівних 20-футовому контейнеру, а другий – в одиницях вимірювання, рівних 40-футовому контейнеру [71].

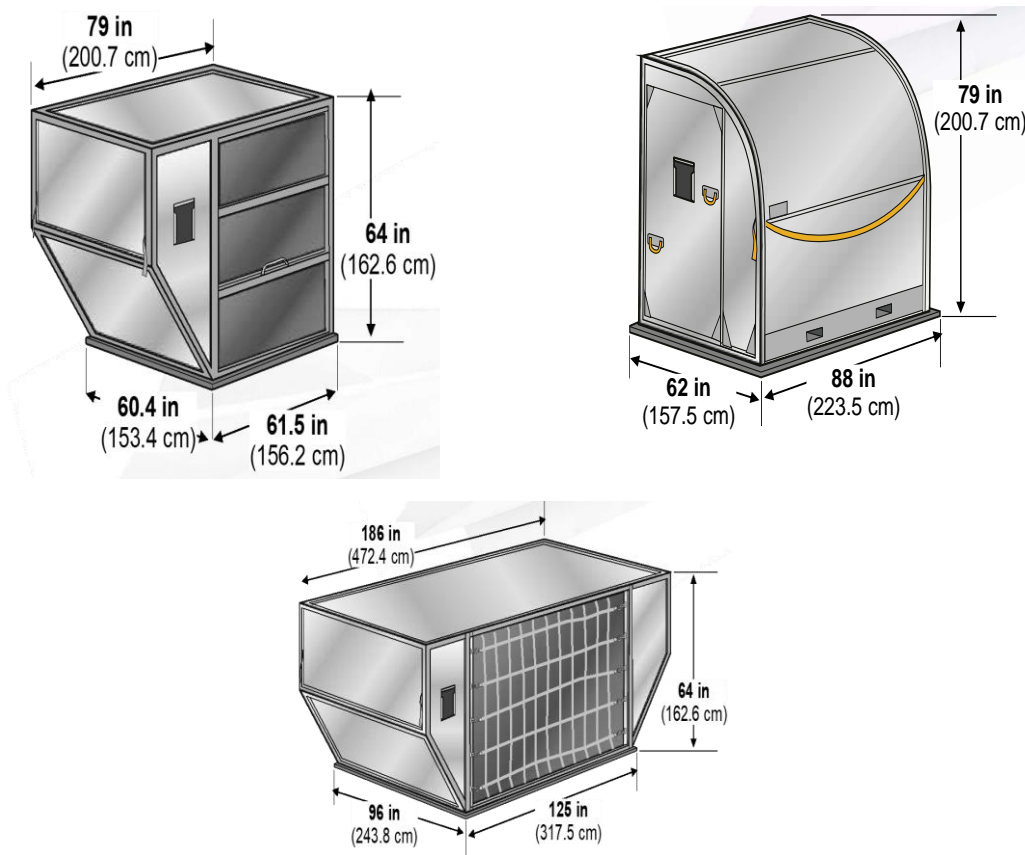
Здебільшого для перевезення вантажів використовують стандартні контейнери у формі прямокутних паралелепіпедів. Однак для низки специфічних видів вантажів, для корпусів транспортних засобів, які мають складну геометричну форму (літаки, підводні човни тощо), застосовують контейнери іншої форми.

Танк-контейнер являє собою ємність з нержавійної сталі об'ємом 12000 – 35000 літрів, у металевій рамі (рис. 1.5), що відповідає розміру ISO 20- або 30-футового контейнера [71].

У великих літаках Boeing вантаж завантажують і перевозять на різних піддонах і контейнерах, відомих як Unit Load Devices (ULD). Зображення і параметри деяких з них наведено на рис. 1.6 [72].

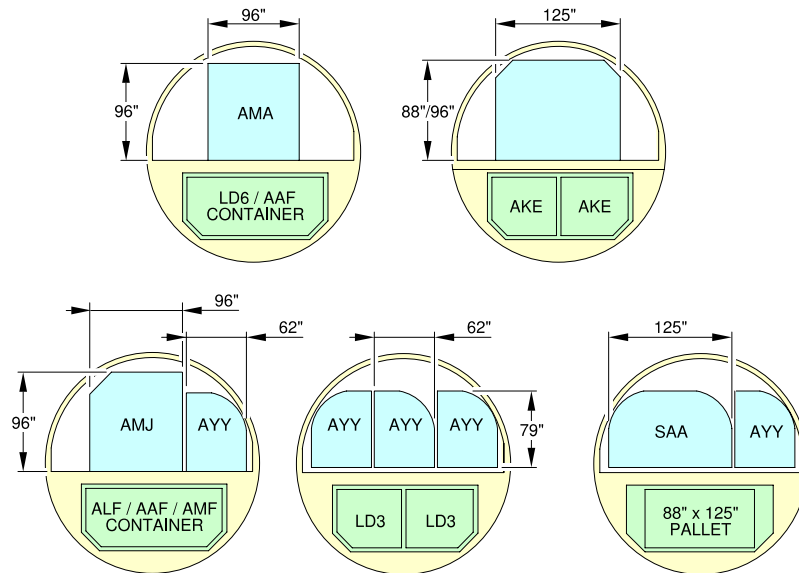


**Рис. 1.5.** Танк-контейнер [71]



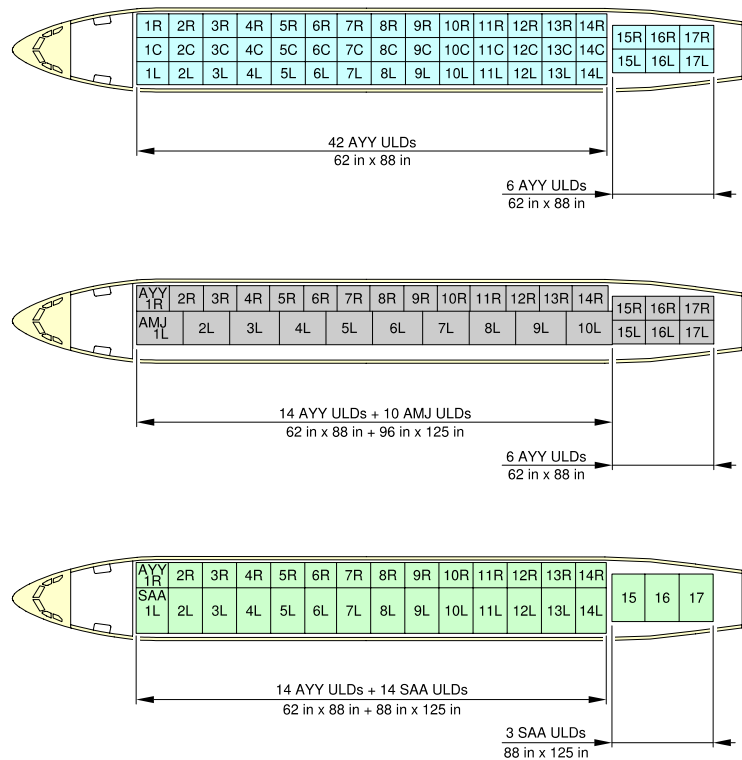
**Рис. 1.6.** Контейнери для літаків Boeing: LD-3, Demi та LD-39 [72]

Розміщення контейнерів на головній та нижній палубах транспортного варіанта літака А-330 показано на рис. 1.7 [73].



**Рис. 1.7.** Розміщення контейнерів на палубах транспортного літака А-330 [73]

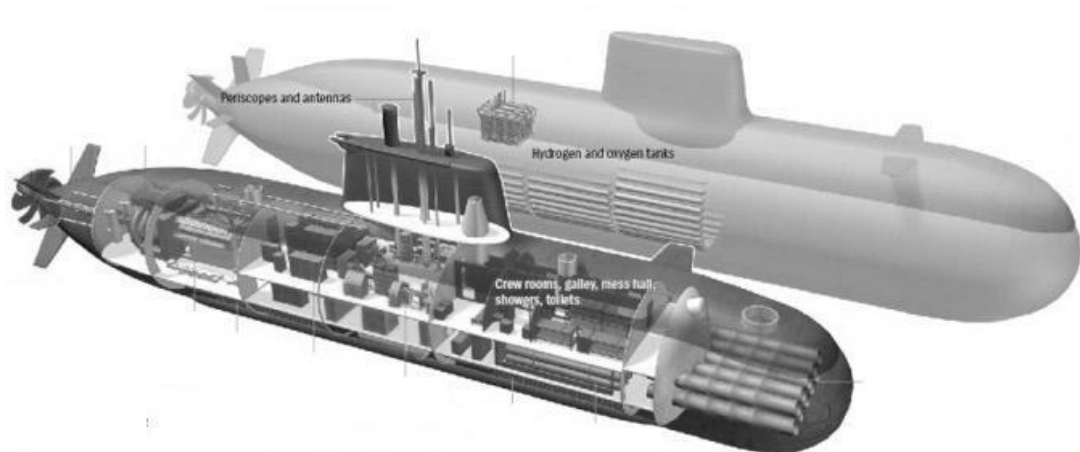
Вантажні відсіки вантажного відділення основної палуби А-330 та схему розміщення контейнерів у них наведено на рис. 1.8 [73].



**Рис. 1.8.** Схема розміщення контейнерів у транспортному А-330 [73]

Можна виокремити клас задач проектування, в яких здійснюється компонування корпусів складної форми об'єктами, що мають різні

геометричні та механічні характеристики. До цього класу належать задачі проєктування морських суден та підводних човнів (рис.1.9),



**Рис. 1.9.** Компонування корпусу підводного човна

компоновання ракет та штучних супутників (рис. 1.10) [74] у ракетно-космічному машинобудуванні.



**Рис. 1.10.** Компонування корпусів ракетно-космічної техніки [74]

Задачі зазначеного класу можна зарахувати до задач проєктування оптимального розміщення корисних вантажів заданих форм та розмірів у контейнерах з урахуванням геометричних та механічних обмежень.

### 1.3. Математичне моделювання та розв'язання задач оптимального завантаження контейнерів

Задачі оптимального завантаження контейнерів, проаналізовані в п. 1.1 та 1.2, є об'єктом системного аналізу, математичного моделювання та застосування методів оптимізації. Наведемо огляд основних результатів щодо оптимізації обслуговування контейнерів на контейнерних терміналах та оптимального розміщення корисних вантажів різних форм у контейнерах довільної форми з урахуванням геометричних та механічних обмежень.

Результати наукових досліджень з системного аналізу, математичного моделювання та оптимізації інтермодальних контейнерних перевезень наведено в працях [75–82].

Одним із важливих класів задач інтермодальних контейнерних перевезень є задачі оброблення контейнерів на залізничних контейнерних терміналах. Огляд задач цього класу наведено в праці [83], що містить загальний опис проблеми та її специфічні випадки.

У класі задач оброблення контейнерів на залізничних контейнерних терміналах однією з головних є проблема складання розкладу руху вантажних потягів та оброблення вантажів на терміналі (*transshipment yards scheduling problem*, TYSP). Цю проблему вперше описано в [84], де сформовано п'ять рівнів вирішення проблеми TYSP.

У статті [84] проблема аналізується на першому рівні, здійснюється формування сервісних слотів потягів, які прибувають на термінал. Побудовано математичну модель та алгоритми розв'язання задачі. Серед них точний алгоритм динамічного програмування та евристичний алгоритм, який використовує евристику променевого пошуку (*beam search*).

В інших працях вдосконалено математичну модель і покращено алгоритми розв'язання. У статті [85] початкова задача TYSP доповнена додатковими обмеженнями, наведено нові алгоритми її розв'язання. У статті [86] наведено удосконалений порівняно з [85] алгоритм гілок та меж за

рахунок більш ефективної нижньої межі Лагранжа. Запропоновано методи отримання двох нових нижніх меж оптимального значення цільової функції.

Згідно з [65] в останні роки моделювання стало важливим інструментом для поліпшення роботи контейнерних терміналів. Виокремлюють три типи моделювання: стратегічне, оперативне та тактичне.

Стратегічне моделювання застосовується для вивчення та порівняння різних типів термінального планування та обладнання для транспортування щодо ефективності та очікуваних витрат. Переважно воно використовується, якщо плануються нові термінали чи якщо необхідно змінити схему або обладнання існуючих терміналів.

Операційне моделювання застосовується для тестування різних видів логістики терміналів та методів оптимізації. Воно використовується здебільшого на великих терміналах. Експлуатація та логістика на великих терміналах дуже складні, і ефект альтернативних методів логістики або оптимізації має бути перевірений об'єктивними методами. Тому методи оптимізації перевіряються в середовищі імітації, перш ніж їх застосовуватимуть у реальних системах керування терміналами та рульового керування.

Тактичне моделювання означає інтеграцію модельних систем в операційну систему терміналу. Варіанти експлуатації моделюються паралельно операції, розробляються рекомендації щодо використання альтернативних варіантів. Реальні дані про роботу терміналу мають імпортуватися та аналізуватися синхронно до операції. Через цю вимогу тактичне моделювання рідко або лише частково використовується на контейнерних терміналах [65].

Задача завантаження контейнера належить до класу задач розміщення геометричних об'єктів, тобто до задач геометричного проектування.

У монографії наведено результати, які є продовженням досліджень, виконаних у відділі математичного моделювання та оптимального проектування Інституту проблем машинобудування ім. А.М. Підгорного НАН

України в рамках теорії геометричного проектування під керівництвом члена-кореспондента НАН України Ю. Г. Стояна. Історію створення та напрями розвитку теорії геометричного проектування розглянуто в [22].

У рамках теорії геометричного проектування для аналітичного опису взаємодії геометричних об'єктів у науковій школі Ю. Г. Стояна розроблено метод  $\phi$ -функцій.

Теорія  $\phi$ -функцій набула суттєвого розвитку для 2D-об'єктів [87–89] і 3D-об'єктів [90–94]. На базі створених засобів математичного моделювання відношень між геометричними об'єктами розроблено математичні моделі та методи розв'язання 2D та 3D оптимізаційних задач розміщення об'єктів [95–99].

Етапом розвитку теорії геометричного проектування стало введення концепції квазі- $\phi$ -функції [100]. Використання квазі- $\phi$ -функцій зумовлює спрощення математичної моделі та формалізацію відношень об'єктів, для яких неможливо побудувати  $\phi$ -функції.

Широке застосування в різних галузях людської діяльності мають задачі пакування та розкрою (Cutting and Packing (C&P)) [101–103]. Вони є NP-складними комбінаторними задачами, тобто це геометричні задачі, в яких об'єкти необхідно розмістити в зоні розміщення так, щоб цільова функція досягала максимального або мінімального значення. Задачі пакування та розкрою належать до задач умовної оптимізації, в кожній з яких необхідно дотримуватися не менше двох умов: 1) усі об'єкти мають міститися всередині заданої зони, 2) об'єкти не мають перетинатися один з одним. Зазначені умови подають у вигляді обмежень розміщення (arrangement constraints) задач C&P.

Різновиди задач C&P трапляються під такими назвами: прямокутний розкрій (Stock-Cutting Problem), пакування в смузі (Strip-Packing Problem), пакування об'єктів у задану зону (Bin-Packing Problem), пакування в рюкзак (Knapsack Problem), задача компоновання (Layout Problem), задача сортування (Assortment Problem), задача вкладення (Nesting Problem), завантаження контейнера (Container Loading) тощо [101–103].

Важливими з практичної точки зору є інженерні застосування задач пакування (серед них Container Loading, Vehicle Loading, Layout problem), які виникають під час компоунування та завантаження сухопутного, водного та повітряного транспорту: кораблів, підводних човнів, потягів, вантажних автомобілів, транспортних літаків, космічних ракет, космічних апаратів тощо. Зазначені задачі належать до класу задач тривимірного пакування (3D-Packing Problem) [103]. Їх можна охарактеризувати множиною обмежень розміщення та додатковими обмеженнями, серед яких пріоритет розміщуваних об'єктів, орієнтація об'єктів, обмеження ваги, рівноваги, стійкості тощо.

Серед напрямів сучасних досліджень з пакування виокремимо задачі балансного компоунування (Balance Layout Problems, BLP). Їх називають також задачами розміщення з додатковими обмеженнями (Constraint Layout Problems). Додатковими обмеженнями в них є обмеження поведінки (behavior constraints), які стосуються технічних характеристик системи (performance constraints) [103, 104].

Задачі балансного компоунування не є достатньо дослідженими. Задача BLP полягає в оптимальному розміщенні набору 3D-об'єктів у деякій обмеженій зоні (контейнері) з метою оптимізації витрат ресурсів у разі дотримання обмежень щодо технічних властивостей системи (рівноваги, стійкості, вібрації тощо). Важливе значення ця задача має для проєктування ракетно-космічної техніки. Проєктований космічний апарат має бути компактним, тобто виготовленим з найменшими витратами дорогих матеріалів. Важливою вимогою до космічних апаратів є виконання поставлених перед ними задач із вказаною точністю. Для цього на початковому етапі проєктування космічного апарата (а саме на етапі компоунування) необхідно враховувати обмеження за статичними та динамічними характеристиками (центр мас, осьові і відцентрові моменти інерції), які впливають на точність стабілізації космічного апарата.

Побудову спрощеної схеми супутникового модуля на базі розв'язання задачі балансного компоунування досліджено в [105]. Для отримання розв'язків

цих задач, зазвичай, використовують евристичні алгоритми, що призводить до втрати локально-оптимальних розв'язків, тоді як у разі застосування методів нелінійної оптимізації час розв'язання задачі є досить великим.

У праці [74] запропоновано сучасні підходи до розв'язання задач пакування в космічному машинобудуванні. Підходи ґрунтуються на апроксимації розміщуваних 3D-об'єктів паралелепіедами та використанні MLP-алгоритмів. Розв'язано задачу оптимального розміщення неорієнтованих об'єктів у параболічному контейнері на підставі мінімізації відхилення центра мас об'єктів від заданої точки.

Згідно з аналізом публікацій [106 –108] можна зробити висновок, що в багатьох задачах пакування як контейнер, зазвичай, використовується прямий коловий циліндр, а як об'єкт розміщення – циліндри або паралелепіеди. Інша ситуація має місце в ракетно-космічному машинобудуванні; використовуються різні форми контейнерів (циліндри, параболоїд обертання, зрізані конуси тощо) та об'єктів (кулі, циліндри, паралелепіеди, правильні призми, тори, сфероциліндри тощо) [74]. Отже, постає потреба у створенні адекватних математичних моделей задач завантаження контейнерів із заданими геометричними формами об'єктів і контейнерів, а також у знаходженні розв'язків, близьких до оптимальних. Для побудови таких математичних моделей у вигляді задач нелінійного програмування є актуальним аналітичний опис вище наведених обмежень.

Ефективним засобом математичного моделювання відношень геометричних об'єктів під час розв'язання задач розміщення є метод  $\rho$ -функцій Стояна. Для розв'язання оптимізаційних задач розміщення метод  $\rho$ -функцій дає змогу використовувати негладку оптимізацію та нелінійне програмування. Перевагою методу  $\rho$ -функцій є можливість урахування мінімально і максимально допустимих відстаней для опису обмежень розміщення об'єктів, що часто потрібно у разі проєктування компонування об'єктів.

Багато задач геометричного проєктування мають комбінаторну структуру та їх описують дискретні оптимізаційні моделі.

Комбінаторна структура задач геометричного проєктування в багатьох випадках залежить від того, що в процесі їх розв'язання необхідно впорядкувати розміщувані об'єкти в заданій зоні. При цьому порядок розміщення визначально впливає на значення критерія ефективності, а з кожним розміщуваним об'єктом пов'язаний набір метричних характеристик і параметрів розміщення. Згідно з цим множини допустимих розв'язків задач геометричного проєктування описують комбінаторними множинами, що належать до різних класів [109, 110].

Математичні моделі багатьох комбінаторних екстремальних задач геометричного проєктування можна побудувати на основі відомих і достатньо досліджених комбінаторних множин, таких як перестановки, розміщення, сполучення та ін. [19, 35, 111, 112]. У той самий час, структура низки задач геометричного проєктування є досить складною і не може бути адекватно описана на основі комбінаторних множин, що належать відомим класам [113, 114].

Задачі оптимального завантаження контейнерів мають складні комбінаторні властивості. Ці властивості зумовлені необхідністю врахування порядку формування слотів обслуговування вантажних потягів, послідовності перевантаження контейнерів з потягів-джерел на цільові потяги та в зону зберігання, вибору вільних платформ та місць у зоні зберігання для завантаження в процесі оптимізації роботи контейнерного терміналу. Для оптимізації завантаження контейнера слід брати до уваги послідовність розміщення об'єктів, варіанти групування розміщуваних об'єктів у разі групового рівноважного компонування, поділ множини розміщуваних об'єктів між відсіками контейнера (підконтейнерами) за їх наявності тощо.

Для побудови математичних моделей наведених задач оптимального завантаження контейнерів запропоновано використовувати комбінаторні конфігурації [26, 30, 33, 109, 110]. Це дає змогу будувати адекватні

математичні моделі задач, що враховують їхні комбінаторні властивості. Методи аналізу побудованих моделей дають можливість отримати ефективні розв'язки задач оптимального завантаження контейнерів.

Стратегії розв'язання дискретно-неперервних задач оптимізації завантаження контейнерів базуються на комплексному використанні методів дискретної та неперервної оптимізації, точних, наближених та евристичних методів. Використовуються методи недиференційовної оптимізації, стохастичні методи оптимізації, субградієнтні методи для лінійних та нелінійних моделей оптимізації, реалізовані в сучасних солверах [1, 4, 26, 74, 115].

Особливу увагу привертає розв'язання задач пакування з інженерним застосуванням, таких як *Layout problems*, *Container Loading* і *Vehicle Loading* [162], в яких вирішуються проблеми компоновання і навантаження наземного, морського і повітряного транспорту: вантажних машин, кораблів, підводних човнів, літаків, ракет, космічних кораблів, супутників тощо. Такі задачі належать до тривимірних задач розміщення (або *3D-Packing Problems*) [163–167] і відрізняються від основного класу *C & P* тим, що в них розглядається низка практичних обмежень, наприклад [168]: вагові обмеження (заповнення контейнера доти, доки вага всієї системи не перевищить заданого допустимого значення); обмеження рівноважного навантаження (рівномірний розподіл вантажу всередині контейнера); обмеження орієнтації (вертикальна або горизонтальна орієнтація об'єктів розміщення у просторі); обмеження стійкості (якщо не враховувати обмеження стійкості, це може призвести до пошкодження вантажу під час його транспортування, або під час процесу завантаження / розвантаження контейнера) і багато інших.

У межах нашого дослідження вивчено задачі компоновання (*Layout Problems*) з урахуванням обмежень, що забезпечують баланс (рівновага, стійкість та інертність) даної системи. У зарубіжній літературі вони мають назви «*Layout Optimization Problems*» (оптимізаційні задачі компоновання), «*Constrained Layout Optimization Problems*» (оптимізаційні задачі

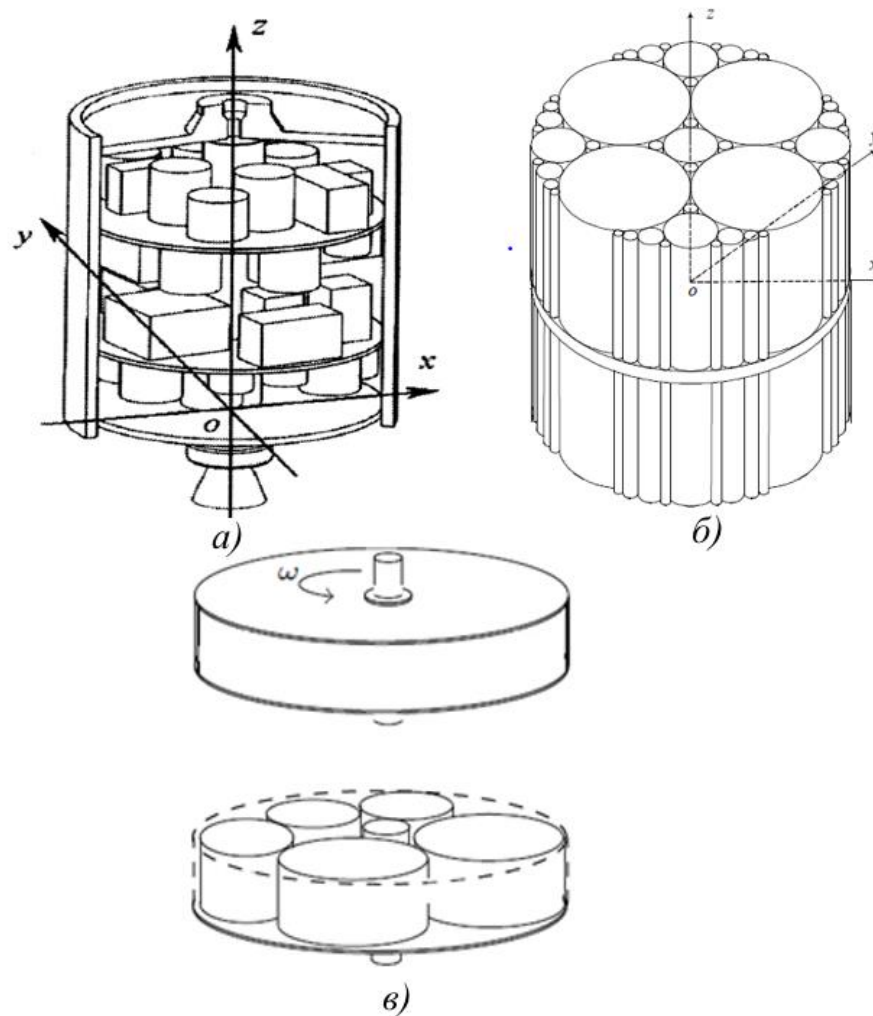
компонування з обмеженнями), «Packing Problems with Constraints» (задачі пакування з обмеженнями), «Packing Problems with Behavior Constraints» (задачі пакування з обмеженнями поведінки) та ін. У монографії для стислості задачі компонування з обмеженнями поведінки визначено як задачі балансного компонування або задачі BLP (Balance Layout Problems).

На сучасному етапі задачі балансного компонування інтенсивно досліджуються. Задачу «Layout Optimization Problem» сформульовано в 1994 році китайськими вченими Н. Teng, S. Sun, W. Ge, W. Zhong у праці «Layout optimization for dishes installed on a rotating table - the packing problem with equilibrium behavioral constraints» [169]. Зазначимо, що з того часу чимало публікацій за цією тематикою належить саме китайській науковій школі Даляньського технологічного університету, представниками якої є такі вчені як Hong-fei Teng, Zhi-Guo Sun, Yi-shou Wang, Chao Che, Yu Chen і багато інших.

Однією з передових публікацій з цього напрямку є праця Z. G. Sun & H. F. Teng [170]. Автори розглядають спрощену модель супутникового модуля, що являє собою контейнер з опорними стелажми (bearing plates) (корпус космічного апарата) і компоновані об'єкти (обладнання), які фіксуються на стелажках всередині контейнера. Як приклад супутникової системи обрано Міжнародний супутник зв'язку Європейського космічного агентства. Його корпус має циліндричну форму, а як обладнання розглядають однорідні об'єкти – циліндри і паралелепіпеди (рис. 1.11, а). Задача полягає в такому: необхідно розмістити вказану кількість об'єктів на кругових стелажках контейнера так, щоб моменти інерції супутникової системи були мінімальними (об'єкти перебували якомога ближче до центра мас системи) у разі дотримання таких обмежень: неперетинання об'єктів і належність об'єктів контейнеру, обмежень на положення центра мас системи і на орієнтацію системи в просторі. Для розв'язання поставленої задачі запропоновано дворівневу стратегію: побудову початкового розміщення (initial layout construction) і пошук оптимального розміщення (layout optimization). На

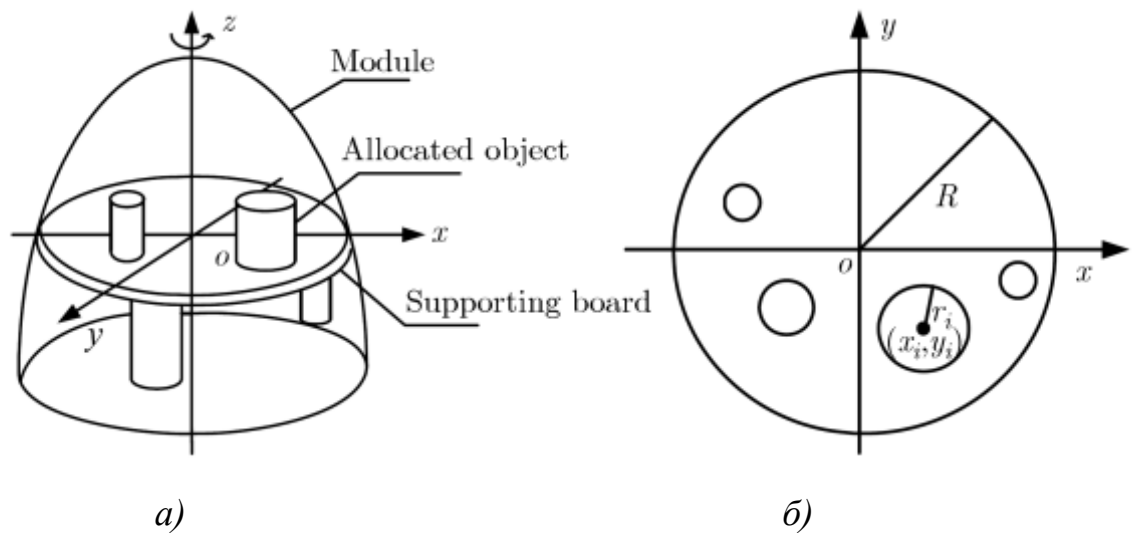
першому етапі здійснюється розподіл множини об'єктів на підмножини відповідно до набору евристичних правил, який називається *centripetal balancing method* (метод доцентрового рівноважного компонування). На другому етапі знаходять наближені розв'язки квазізавдання комівояжера (а *quasi Traveling Salesman Problem*) за допомогою мурашиного алгоритму (*Ant Colony Optimization*) – метаевристичного алгоритму оптимізації наслідуванням мурашиної колонії.

У працях [171,172] також досліджено задачі, які знаходять застосування для компонування супутникового модуля. Розглянуто задачі оптимального розміщення 3D-об'єктів у циліндричному контейнері з круговими стелажми. Наведено математичні моделі з обмеженнями розміщення і обмеженнями поведінки, де функція цілі – радіус основи контейнера. Головною метою цих публікацій є опис тестових завдань з наперед відомими наперед оптимальними розв'язками для оцінювання ефективності запропонованих алгоритмів. Звернено увагу, що оцінити ефективність того чи іншого алгоритму можна трьома способами: складним теоретичним аналізом, перевіркою алгоритму на практиці за допомогою інженерного тестування, а також за допомогою тестового завдання, оптимальне значення якого наперед відомо. Для виконання умов поведінки системи тестові завдання побудовано з використанням властивостей симетрії супутникової системи (рис. 1.11, б).



**Рис. 1.11.** Спрощені моделі супутникових систем

Проблеми аналізу оптимізаційних задач балансного компонування також розглянуто в працях китайських вчених Liu JingFa і Li Gang з Нанкінського політехнічного університету (Nanjing University of Information Science & Technology) [173]. Їхні дослідження присвячені компонуванню кругових об'єктів у кругову зону з урахуванням обмежень рівноваги (equilibrium behavioral constraints) (рис. 1.12). Для розв'язання такої NP-важкої задачі запропоновано алгоритм глобального пошуку (global search algorithm) під назвою алгоритм заповнення резервуара (basin filling algorithm), який поєднує в собі вдосконалений метод the energy landscape paving (ELP), метод градієнтів, заснований на пошуку локальних екстремумів, і евристичний механізм оновлення конфігурації (heuristic configuration update mechanism).



**Рис.1.12.** Схематичне зображення розміщення об'єктів у модульному відсіку супутника: а – 3D вид системи, б – 2D вид системи

У праці [174] запропоновано алгоритм для розв'язання такого завдання балансного компоновання. Йдеться про розміщення кругових об'єктів, наділених масами, в круговому контейнері з метою мінімізації радіуса контейнера і невірноваженості системи. Отже, розглянуто функцію цілі у вигляді згортки двох функцій і враховано обмеження неперетину об'єктів, розташованих у заданій зоні. Запропоновано генетичний алгоритм, який порівнюється з двома іншими алгоритмами, «натхненними» природою: модифікованим генетичним алгоритмом і алгоритмом, заснованим на методі рою частинок (Particle Swarm Optimizer).

Китайським вченим Kaiyou Lei з Південно-Західного університету Чунцина у праці [175] розглянуто двовимірну оптимізаційну задачу розміщення об'єктів з урахуванням поведінкових обмежень. Задача полягає в такому: необхідно розмістити круглі однорідні диски, наділені масами, в круглому контейнері, що обертається (див. рис. 1.11, в). Диски мають розміщуватися якомога ближче до центра контейнера у разі виконання обмежень неперетину дисків, входження дисків до заданої зони та обмежень динамічного балансування системи. Як цільову функцію вибрано радіус контейнера (охоплювального кола). Для розв'язання цієї задачі запропоновано

адаптивний метод рою частинок (Adaptive Particle Swarm Optimizer). Наведено результати тестових прикладів, а також виконано порівняльний аналіз з іншими працями.

Аналогічну задачу досліджено бразильськими вченими W.A. Oliveira, L.L. Salles Neto et al. у праці [176]. Розглянуто двокритеріальну задачу: як функцію цілі запропоновано дві величини: радіус контейнера і функцію, яка вимірює динамічну неврівноваженість системи, зумовлену обертанням контейнера. Для розв'язання задачі запропоновано евристичний алгоритм під назвою «center-of-mass-based placing technique», який полягає в розміщенні кіл навколо центра мас системи і переміщенні початку системи координат у разі кожного додавання кола в поточний стан центра мас системи з метою збереження рівноваги системи. Доведено, що запропонований алгоритм показав кращий результат, ніж алгоритми, наведені в [175, 177, 178].

Огляд наукових публікацій показав, що для розв'язання задач балансного компонування використовуються переважно евристики і мета-евристики (різні модифіковані методи оптимізації з використанням евристичних алгоритмів). Найбільш популярними є метод рою частинок (PSO – Particle swarm optimization) [175, 177–182], метод імітації відпалу (SA – simulated Annealing) [178], різні модифікації генетичного алгоритму (GA – Genetic Algorithm) [174, 183–186], мурашиний алгоритм (ACO – Ant Colony Optimization) [170, 187, 188] та інші оптимізаційні процедури [176, 189–194].

Вагомий внесок у розвиток методів розв'язання оптимізаційних задач з ракетно-космічного машинобудування зробили такі вчені, як Giorgio Fasano та Janos D. Pinter. Giorgio Fasano – італійський дослідник і практик з математичного моделювання та оптимізації, що працює у великій європейській компанії «Thales Alenia Space» [<https://www.thalesgroup.com/en>]. Ця компанія спеціалізується на розробленні, побудові, тестуванні та керуванні інноваційними космічними системами. Деякі системи, створені в «Thales Alenia Space», наведено на рис. 1.13. Janos D. Pinter – канадський дослідник і

практик з моделювання та оптимізації систем (systems modeling and optimization), зокрема, з розроблення алгоритмів і програмного забезпечення.



*a)*

*б)*

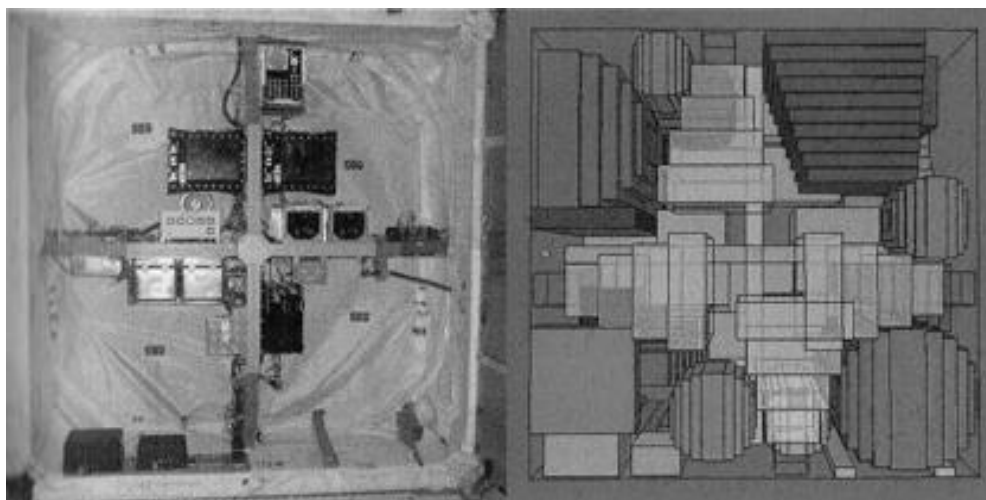
**Рис.1.13.** Космічні апарати, розроблені в «Thales Alenia Space»:

*a* – «StratoBus» – гібрид супутника і безпілотної літака, *б* – метеорологічний супутник «MSG-4»

У виданнях серії Springer Optimization, опублікованих за редакцією G. Fasano і J. Pintér, наведено низку праць [195–203], присвячених оптимізаційним задачам пакування в ракетно-космічному машинобудуванні. Серед сучасних підходів до розв’язання задач пакування в космічному машинобудуванні – MILP-алгоритми (див., наприклад, [196, 200, 202] – Mixed Integer Linear Programming Algorithms – алгоритми цілочислового лінійного програмування), для застосування яких використовується апроксимація розміщуваних 3D-об’єктів  $\delta$ -паралелепіпедами, як показано на рис. 1.14.

Зазначимо, що задачі розміщення геометричних об’єктів з урахуванням їх центрів ваги досліджували в українській науковій школі Ю. Г. Стояна (див., наприклад, [204]). Задача полягала в пошуку мінімального відхилення центра ваги системи розміщуваних 2D-об’єктів від заданої точки. При цьому враховували зони заборони в заданій зоні розміщення, мінімально допустимі відстані між кожною парою об’єктів, між кожним об’єктом і зоною заборони,

а також між об'єктом і зовнішньою межею зони розміщення. На базі годографа вектор-функції щільного розміщення [205], а також функції щільного розміщення сформульовано точну математичну постановку задач оптимального розміщення об'єктів. Для розв'язання розглянутих задач запропоновано методи із застосуванням апарата теорії статистичних розв'язків: метод випадкового пошуку за рівноймовірної вибірки початкових точок з подальшим отриманням локальних екстремумів і метод асимптотичного перебору локальних екстремумів. Розглянуто конкретні приклади розміщення 40 прямокутних об'єктів у восьмикутній зоні розміщення. Подібні задачі знаходять застосування в суднобудуванні, авіації, ракетобудуванні тощо.



**Рис. 1.14.** Апроксимація обладнання, що розміщується в лабораторному модулі «Каламбус» Міжнародної космічної станції

У книзі Стояна Ю. Г., Куліша Е. Н. [206] розглянуто задачу компоновання бортового обладнання літальних апаратів, викладено математичні методи, що дають змогу формалізувати і розв'язати такі завдання за допомогою ЕОМ. Ефективність методів проілюстровано різними тестовими прикладами. У праці зазначено, що проектування компоновальних схем обладнання літальних апаратів виконується переважно евристично:

способами, заснованими на традиціях, інтуїції і досвіді розробників. При цьому трудовитрати на компоновальні роботи великі, а варіанти розміщення часто далекі від «найкращих».

Один з розділів монографії [198], написаний Yu. Stoyan & T. Romanova, присвячений математичному моделюванню задач розміщення. У ньому розглянуто рівноважне пакування циліндрів і паралелепіпедів у параболічному контейнері, розділеному круговими стелажми. Наведено математичну модель задачі з використанням методу  $\phi$ -функцій. Для розв'язання задачі використовуються методи нелінійної оптимізації, які дають змогу знаходити локально оптимальні розв'язки.

У статті [207] автори P. Stetsyuk, T. Romanova, G. Scheithauer розглядають задачу балансного компоновання кіл у коло мінімального радіуса, що являє собою багатоекстремальну задачу нелінійного програмування. За допомогою негладких штрафів задачу зводять до задачі безумовної мінімізації негладкої функції. Запропоновано ефективний алгоритм пошуку локальних екстремумів, а також алгоритм, який поліпшує нижню межу глобального мінімуму функції цілі. Ці алгоритми використовують методи негладкої оптимізації, засновані на  $r$ -алгоритмі Шора [208].

#### 1.4. Постановка задачі балансного компоновання (BLP)

*Контейнери.* Нехай  $\Omega$  – контейнер висотою  $H$ , який описують так:  $\Omega = \{(x, y, z) \in \mathbb{R}^3 : G(x, y, z) \geq 0\}$ . Контейнер  $\Omega$  розділений круговими стелажми  $S_k$ ,  $k = 1, 2, \dots, m$ , на підконтейнери  $\Omega^k$ ,  $k = 1, 2, \dots, m$ . Позначимо  $S_1$  і  $S_{m+1}$  нижню та верхню основи контейнера  $\Omega$  відповідно. Нехай  $t_k$  – задана відстань між стелажми  $S_k$  і  $S_{k+1}$ . Позначимо  $Oxyz$  власну систему координат контейнера  $\Omega$ , початок  $O$  якої знаходиться в центрі симетрії

нижньої основи  $S_1$  контейнера  $\Omega$ ,  $Oz$  – поздовжня вісь симетрії контейнера  $\Omega$  (рис. 1.15).

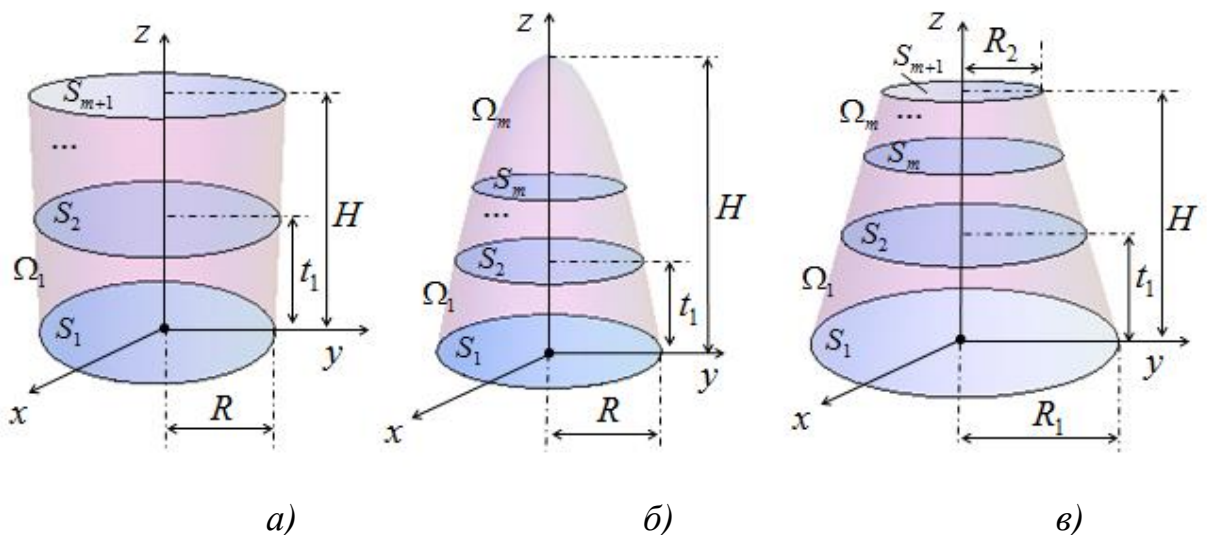
Розглядають такі види контейнерів  $\Omega$ :

1)  $\Omega \equiv \mathbf{C}$ , де  $\mathbf{C}$  – прямий круговий циліндр з основами радіусом  $R$ ,  
 $G(x, y, z) = \min\{-x^2 - y^2 + R^2, -z + H, z\}$  (рис.1.15, а);

2)  $\Omega \equiv \mathbf{\Lambda}$ , де  $\mathbf{\Lambda}$  – параболоїд обертання з основою радіусом  $R = \sqrt{H}$ ,  
 $G(x, y, z) = \min\{-z - x^2 - y^2 + H, z\}$  (рис.1.15, б);

3)  $\Omega \equiv \mathbf{E}$ ,  $\mathbf{E}$  – прямий коловий зрізаний конус із радіусами  $R_1$  і  $R_2$  нижньої і верхньої основ відповідно, ( $R_1 > R_2$ ), для якого  
 $G(x, y, z) = \min\{-z - H \cdot (\sqrt{x^2 + y^2} + R_1) / (R_1 - R_2), -z + H, z\}$  (рис.1.15, в), або  
 $G(x, y, z) = \min\{-z - H_1 \cdot (\sqrt{x^2 + y^2} + R_1) / R_1, -z + H, z\}$ , де  $H_1$  – висота конуса, який генерує зрізаний конус  $\mathbf{E}$ ,  $H_1 = HR_1 / (R_1 - R_2)$ .

Контейнер  $\Omega$  розглядають як однорідну поверхню масою  $m_0$ . Масою основ і стелажів  $S_k$ ,  $k = 1, 2, \dots, m+1$ , контейнера  $\Omega$  нехтуємо.



**Рис. 1.15.** Види контейнерів: а – циліндр, б – параболоїд обертання, в – зрізаний конус

Множина об'єктів  $A = \{A_i, i \in I_n\}$ ,  $I_n = \{1, 2, \dots, n\}$ , включає:

- 1) кулі  $S_i$  радіусів  $r_i$ ,  $i \in I_1 = \{1, 2, \dots, n_1\}$ ;
- 2) прямі колові циліндри  $C_i$  радіусами  $r_i$  та висотами  $2h_i$ ,  
 $i \in I_2 = \{n_1 + 1, \dots, n_1 + n_2\}$ ;
- 3) тори  $T_i$  з метричними характеристиками  $(r_i, h_i)$ , де  $r_i$  – відстань від центра твірного кола до осі обертання,  $2h_i$  – висота тора,  
 $i \in I_3 = \{n_1 + n_2 + 1, \dots, n_1 + n_2 + n_3\}$ ;
- 4) сфероциліндри  $S_{C_i}$  з метричними характеристиками  $(l_i, r_i, h_i)$ , де  $l_i$  – висоти кульових сегментів,  $r_i$  – радіус циліндра,  $2h_i$  – висота циліндра,  
 $i \in I_4 = \{n_1 + n_2 + n_3 + 1, \dots, n_1 + n_2 + n_3 + n_4\}$ ;
- 5) прямі прямокутні паралелепіпеди  $P_i$  з метричними характеристиками  $(w_i, l_i, h_i)$ , де  $2w_i$  – довжина,  $2l_i$  – ширина,  $2h_i$  – висота паралелепіпеда,  
 $i \in I_5 = \{n_1 + n_2 + n_3 + n_4 + 1, \dots, n_1 + n_2 + n_3 + n_4 + n_5\}$ ;
- б) прямі правильні призми  $K_i$  з метричними характеристиками  $(r_i, h_i)$ , де  $r_i$  – радіус основи циліндра, описаного навколо  $K_i$ ,  $2h_i$  – висота призми,  
 $i \in I_6 = \{n_1 + n_2 + n_3 + n_4 + n_5 + 1, \dots, n\}$ , причому  $I_1 \cup I_2 \cup \dots \cup I_6 = I_n$ . Позначимо  $O_i x_i y_i z_i$  власну систему координат об'єкта  $A_i$ , початок  $O_i$  якої знаходиться в центрі симетрії об'єкта  $A_i$ , де  $O_i x_i$ ,  $O_i y_i$ ,  $O_i z_i$  – вісі симетрії об'єкта  $A_i$  та  $O_i z_i \parallel Oz$  (рис. 1.16). Об'єкти множини  $A$  орієнтовані так, як показано на рис. 1.12. Допускаються неперервні обертання об'єкта навколо осі  $O_i z_i$ . Крім того, кожний об'єкт  $A_i$  являє собою однорідне тверде тіло масою  $m_i$ .

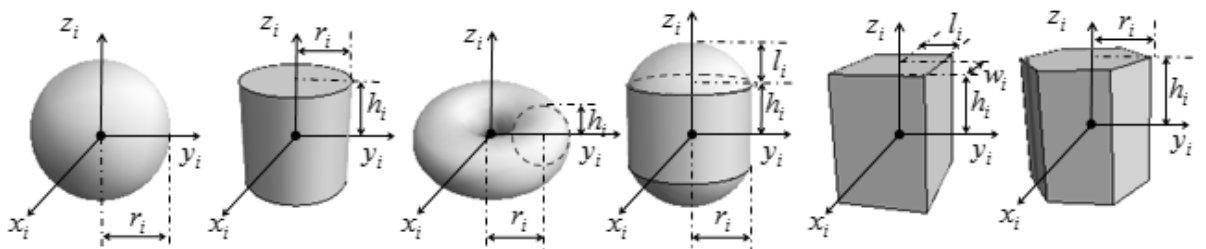


Рис. 1.16. Види об'єктів компонування

Уважаємо, що задано розбиття множини  $A$  об'єктів на підмножини  $A^k = \{A_i, i \in I^k\}$ ,  $I^k = \{1, 2, \dots, n^k\}$ ,  $k = 1, 2, \dots, m$ , згідно з розміщенням об'єктів всередині відповідних підконтейнерів  $\Omega^k$ ,  $k = 1, 2, \dots, m$ , тоді  $I^1 \cup I^2 \cup \dots \cup I^k \cup \dots \cup I^m = I_n$ . У свою чергу, підмножина  $A^k$  об'єктів, у загальному випадку, розбивається на дві підмножини  $A_+^k = \{A_i, i \in I_+^k\}$  і  $A_-^k = \{A_i, i \in I_-^k\}$ , де  $A_+^k$  – підмножина об'єктів, які мають бути розміщені на стелажі  $S_k$ ,  $A_-^k$  – підмножина об'єктів, які мають бути розміщені *нид* стелажем  $S_{k+1}$  всередині підконтейнера  $\Omega^k$ .

Контейнер  $\Omega$  з упакованими в ньому об'єктами множини  $A$  утворює систему  $\Omega_A$ . Задамо систему координат  $O_sXYZ$  для системи  $\Omega_A$ , початок  $O_s$  якої розташований у центрі мас системи  $\Omega_A$ , а вісі  $O_sX, O_sY, O_sZ$  паралельні осям  $Ox, Oy, Oz$  відповідно (рис. 1.17).

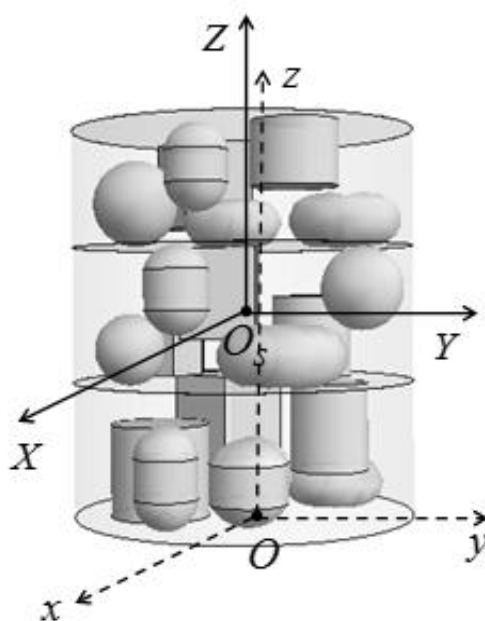


Рис. 1.17. Приклад системи  $\Omega_A$

Розташування об'єктів множини  $A$  всередині контейнера  $\Omega$  визначається, в загальному випадку, вектором параметрів розміщення  $u_i = (x_i, y_i, z_i, \theta_i)$  відносно системи координат  $Oxyz$ , де  $(x_i, y_i, z_i)$  – вектор

трансляції об'єкта  $A_i$ , а  $\theta_i$  – кут повороту об'єкта  $A_i$  в площині  $O_i x_i y_i$  (рис. 1.18). При цьому координата  $z_i$  вектора параметрів розміщення об'єкта  $A_i$  набуває фіксованих значень, які визначають так:

- для  $C_i, T_i, K_i, P_i$ :

$$z_i = \sum_{l=1}^k t_{l-1} + h_i, \text{ якщо } i \in I_+^k, \quad z_i = \sum_{l=1}^k t_l - h_i, \text{ якщо } i \in I_-^k; \quad (1.1)$$

- для  $S_i$ :

$$z_i = \sum_{l=1}^k t_{l-1} + r_i, \text{ якщо } i \in I_+^k, \quad z_i = \sum_{l=1}^k t_l - r_i, \text{ якщо } i \in I_-^k; \quad (1.2)$$

- для  $S_{C_i}$ :

$$z_i = \sum_{l=1}^k t_{l-1} + h_i + l_i, \text{ якщо } i \in I_+^k, \quad z_i = \sum_{l=1}^k t_l - h_i - l_i, \text{ якщо } i \in I_-^k. \quad (1.3)$$

Тоді вектор змінних задачі визначають як  $(p, u) = (p, u_1, u_2, \dots, u_n) \in \mathbb{R}^{3n+1}$ , де  $u_i = (x_i, y_i, z_i, \theta_i)$ ,  $i \in I_n$ ,  $p$  – змінна метрична характеристика контейнера  $\Omega$ .

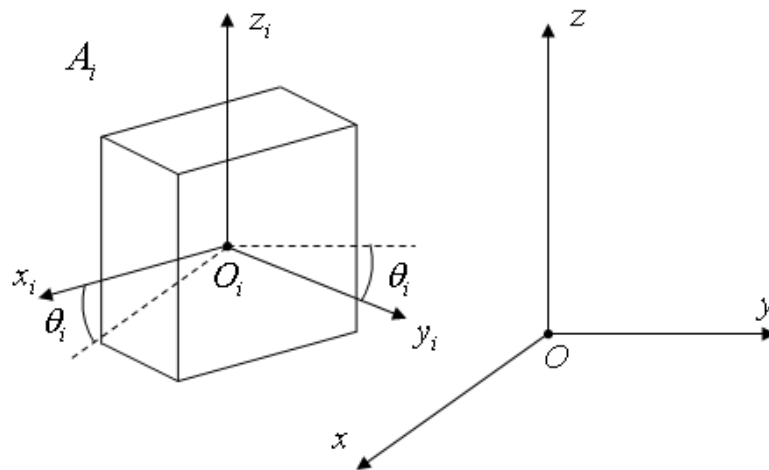


Рис. 1.18. Система координат  $O_i x_i y_i z_i$  відносно  $Oxyz$

Розглядають такі види обмежень задачі:

1. Обмеження розміщення (arrangement constraints), які діляться на:

- *обмеження неперетину (non-overlapping constraints) з урахуванням допустимих відстаней*: об'єкти  $A_i, A_j \in A^k$ ,  $j > i \in I^k$ ,  $k = 1, 2, \dots, m$ , мають бути розміщені не ближче, ніж на мінімально допустиму відстань  $\rho_{ij}^-$ , і не далі, ніж на максимально допустиму відстань  $\rho_{ij}^+$ , тобто  $\rho_{ij}^- \leq \text{dist}(A_i, A_j) \leq \rho_{ij}^+$ ,  $j > i \in I^k$ ,  $k = 1, 2, \dots, m$ ,

- *обмеження включення (containment constraints) з урахуванням допустимих відстаней*: об'єкти  $A_i \in A^k$ ,  $i \in I^k$ , повинні бути включені в підконтейнер  $\Omega^k$ ,  $k = 1, 2, \dots, m$ , з урахуванням заданих мінімально допустимих відстаней  $\rho_i^-$ , тобто  $\text{dist}(A_i, \Omega^{k*}) \geq \rho_i^-$ ,  $i \in I^k$ ,  $\Omega^{k*} = \mathbb{R}^3 \setminus \text{int } \Omega^k$ ,  $k = 1, 2, \dots, m$ ;

2. Обмеження поведінки (behavior constraints), які діляться на:

- *обмеження рівноваги: відхилення центра мас системи  $\Omega_A$  від заданої точки не повинно перевищувати заданого допустимого значення,*

- *обмеження моментів інерції: значення осьових моментів інерції системи  $\Omega_A$  не повинно перевищувати заданого допустимого значення,*

- *обмеження стійкості: значення відцентрових моментів інерції не повинні перевищувати заданих допустимих значень.*

**Задача балансного компоунвання (BLP – Balance Layout Problem):**

розмістити об'єкти підмножин  $A^k$  на стелажах  $S_k$ ,  $k = 1, 2, \dots, m+1$ , контейнера  $\Omega$  з урахуванням обмежень розміщення і обмежень поведінки так, щоб задана функція цілі  $F(p, u)$  досягала свого екстремального значення.

Наведений у розділі 1 аналіз публікацій за даною тематикою показав, що у формуванні функції цілі  $F(p, u)$  задачі балансного компоунвання найчастіше беруть участь метрична характеристика контейнера  $\Omega$  і

відхилення центра мас системи  $\Omega_A$  від заданої точки. Крім того, як цільову функцію  $F(p, u)$  використовують згортку осьових моментів інерції системи  $\Omega_A$  (див., наприклад, [170]).

### 1.5. Основні визначення методу рхі-функцій

Для аналітичного опису обмежень розміщення застосовують метод рхі-функцій [25, 87, 100].

Наведемо основні відомості стосовно методу рхі-функцій, суттєві для подальшого викладення результатів дослідження. Нехай є пара замкнутих рхі-об'єктів  $T_1 \subset R^t$  та  $T_2 \subset R^t$ ,  $t = 2, 3$  [116]. Позначимо  $u_1 = (x_1, y_1, z_1, \theta_1)$  та  $u_2 = (x_2, y_2, z_2, \theta_2)$  вектори змінних об'єктів  $T_1$  та  $T_2$  відповідно, де  $(x_i, y_i, z_i)$  – вектор трансляції об'єкта  $T_i$ , а  $\theta_i = (\theta_{zi}, \theta_{xi}, \theta_{yi})$  – вектор кутів повороту: від осі  $OX$  до  $OY$ , від осі  $OY$  до  $OZ$  та від осі  $OX$  до  $OZ$ ,  $i = 1, 2$ . Надалі об'єкт  $T_i$ ,  $i = 1, 2$ , послідовно обернутий на кути  $\theta_{zi}, \theta_{xi}, \theta_{yi}$  та трансльований на вектор  $(x_i, y_i, z_i)$ , позначатимемо як  $T_i(u_i)$ ,  $i = 1, 2$ .

З метою моделювання відношень неперетину об'єктів і включення об'єктів у зоні розміщення, зокрема, в задачах завантаження контейнерів, використовуються рхі-функції, визначення і властивості яких розглянуто, наприклад, у [25, 100].

*Визначення 1.1.* Всюди визначена неперервна функція  $\Phi(u_1, u_2)$  для пари об'єктів  $T_1(u_1)$  і  $T_2(u_2)$  називається рхі-функцією, якщо виконуються такі три умови:

$$\Phi(u_1, u_2) > 0, \text{ якщо } T_1(u_1) \cap T_2(u_2) = \emptyset;$$

$$\Phi(u_1, u_2) = 0, \text{ якщо } \begin{cases} \text{int } T_1(u_1) \cap \text{int } T_2(u_2) = \emptyset; \\ \text{fr } T_1(u_1) \cap \text{fr } T_2(u_2) \neq \emptyset \end{cases};$$

$$\Phi(u_1, u_2) < 0, \text{ якщо } \text{int } T_1(u_1) \cap \text{int } T_2(u_2) \neq \emptyset.$$

Уважаємо, що початок власної системи координат (полнос  $(x_i, y_i, z_i)$ ) центрально-симетричних об'єктів знаходиться в їх центрі симетрії.

Для опису відношень включення об'єктів у зону (контейнер), тобто  $T_2(u_2) \subset T_1(u_1) \Leftrightarrow \text{int } T_2(u_2) \cap \text{int } T_1^*(u_1) = \emptyset$ , використовуємо властивості  $\phi$ -функції:  $\Phi^{T_1^* T_2}(u_1, u_2) \geq 0$ , якщо  $\text{int } T_2(u_2) \cap \text{int } T_1^*(u_2) = \emptyset$ .

Умову неперетину пари об'єктів  $T_1(u_1)$  і  $T_2(u_2)$  також можна записати за допомогою квазі- $\phi$ -функції, визначення і властивості якої наведено в [25, 100].

*Визначення 1.2.* Квазі- $\phi$ -функцією для  $\phi$ -об'єктів  $T_1(u_1)$  і  $T_2(u_2)$  називається всюди визначена неперервна функція  $\Phi'(u_1, u_2, u')$ , для якої функція  $\max_{u' \in U} \Phi'(u_1, u_2, u')$  є  $\phi$ -функцією для  $T_1(u_1)$  і  $T_2(u_2)$ , де  $u'$  – додаткові змінні, вид множини  $U \subset \mathbb{R}^\zeta$  і розмірність простору  $\mathbb{R}^\zeta$  залежать від форми розміщуваних об'єктів.

Властивості квазі- $\phi$ -функції  $\Phi'(u_1, u_2, u')$ :

- 1)  $\Phi'(u_1, u_2, u') \geq 0 \Rightarrow \text{int } A(u_A) \cap \text{int } B(u_B) = \emptyset$ ;
- 2) функція виду

$$\Phi'(u_1, u_2, u_P) = \min\{\Phi^{T_1 P}(u_1, u_P), \Phi^{T_2 P^*}(u_2, u_P)\}, \quad (1.4)$$

є квазі- $\phi$ -функцією для обмежених об'єктів  $T_1(u_1)$  і  $T_2(u_2)$ , де  $\Phi^{T_1 P}(u_1, u_P)$  –  $\phi$ -функція для об'єктів  $T_1(u_1)$  і  $P(u_P)$ ,  $\Phi^{T_2 P^*}(u_2, u_P)$  –  $\phi$ -функція для об'єктів  $T_2(u_2)$  і  $P^*(u_P) = \mathbb{R}^t \setminus \text{int } P(u_P)$ ,  $t = 2, 3$ ,

- якщо  $T_1, T_2 \subset \mathbb{R}^3$ , то  $P(u_P)$  – відокремлювальна площина виду  $P(u_P) = \{(x, y, z) : \psi_P = \alpha \cdot x + \beta \cdot y + \gamma \cdot z + \mu_P \geq 0\}$ , де  $\alpha = \sin \theta_{yP}$ ,  $\beta = -\sin \theta_{xP} \cdot \cos \theta_{yP}$ ,  $\gamma = \cos \theta_{xP} \cdot \cos \theta_{yP}$ ,  $u_P = (\theta_{xP}, \theta_{yP}, \mu_P)$ ;

• якщо  $T_1, T_2 \subset \mathbb{R}^2$ , то  $P(u_P)$  – відокремлювальна пряма виду  $P(u_P) = \{(x, y) : \psi_P = \alpha \cdot x + \beta \cdot y + \mu_P \geq 0\}$ , де  $\alpha = \cos \theta_P$ ,  $\beta = \sin \theta_P$ ,  $u_P = (\theta_P, \mu_P)$ .

Нехай задано мінімально і максимально допустимі відстані  $\rho^-$  і  $\rho^+$  між phi-об'єктами  $T_1(u_1)$  і  $T_2(u_2)$ , тобто

$$\rho^- \leq \text{dist}(T_1, T_2) \leq \rho^+, \quad (1.5)$$

де  $\text{dist}(T_1, T_2) = \min_{t_1 \in T_1, t_2 \in T_2} d(t_1, t_2)$ ,  $d(t_1, t_2)$  – евклідова відстань між точками  $t_1$  і  $t_2$ .

Обмеження (1.4) можна записати за допомогою псевдонормалізованих phi-функцій, визначення і властивості яких наведено в [25, 100].

*Визначення 1.3.* Всюди визначена неперервна функція  $\hat{\Phi}^-(u_1, u_2)$  (або  $\hat{\Phi}^+(u_1, u_2)$ ) називається псевдонормалізованою phi-функцією для phi-об'єктів  $T_1(u_1)$  і  $T_2(u_2)$ , якщо виконуються такі умови:

$$\begin{aligned} \hat{\Phi}^-(u_1, u_2) &> 0, \text{ якщо } \text{dist}(T_1, T_2) > \rho^-, \\ \hat{\Phi}^+(u_1, u_2) &> 0, \text{ якщо } \text{dist}(T_1, T_2) < \rho^+, \\ \hat{\Phi}^-(u_1, u_2) &= 0, \text{ якщо } \text{dist}(T_1, T_2) = \rho^-, \\ \hat{\Phi}^+(u_1, u_2) &= 0, \text{ якщо } \text{dist}(T_1, T_2) = \rho^+, \\ \hat{\Phi}^-(u_1, u_2) &< 0, \text{ якщо } \text{dist}(T_1, T_2) < \rho^-, \\ \hat{\Phi}^+(u_1, u_2) &< 0, \text{ якщо } \text{dist}(T_1, T_2) > \rho^+. \end{aligned}$$

За допомогою псевдонормалізованих phi-функцій не завжди можна описати розміщення об'єктів з урахуванням їх неперервних обертань і допустимих відстаней. Для опису обмежень на мінімально і максимально допустимі відстані використовуються псевдонормалізовані квазі-phi-функції, визначення і властивості яких наведено в [25, 100].

*Визначення 1.4.* Псевдонормалізованою квазі- $\Phi$ -функцією для об'єктів  $T_1(u_1)$  і  $T_2(u_2)$  називається функція  $\widehat{\Phi}'(u_1, u_2, u')$ , для якої функція  $\max_{u' \in \mathbb{R}^d} \widehat{\Phi}'(u_1, u_2, u')$  є псевдонормалізованою  $\Phi$ -функцією для об'єктів  $T_1(u_1)$  і  $T_2(u_2)$ .

Подібно до псевдонормалізованих  $\Phi$ -функцій введемо псевдонормалізовані квазі- $\Phi$ -функції  $\widehat{\Phi}'^-$  для моделювання обмежень  $\text{dist}(T_1, T_2) \geq \rho^-$  та псевдонормалізовані квазі- $\Phi$ -функції  $\widehat{\Phi}'^+$  для моделювання обмежень  $\text{dist}(T_1, T_2) \leq \rho^+$ .

$$\text{Тоді } \max_{u' \in U} \widehat{\Phi}'^- \geq 0 \Leftrightarrow \text{dist}(T_1, T_2) \geq \rho^-, \quad \max_{u' \in U} \widehat{\Phi}'^+ \geq 0 \Leftrightarrow \text{dist}(T_1, T_2) \leq \rho^+.$$

Використовуючи властивості квазі- $\Phi$ -функції, визначимо властивості псевдонормалізованої квазі- $\Phi$ -функції:

- 1)  $\widehat{\Phi}'^- \geq 0 \Rightarrow \text{dist}(T_1, T_2) \geq \rho^-, \quad \widehat{\Phi}'^+ \geq 0 \Rightarrow \text{dist}(T_1, T_2) \leq \rho^+,$
- 2) функція

$$\widehat{\Phi}'^-(u_1, u_2, u_p) = \Phi'(u_1, u_2, u_p) - 0.5\rho^-,$$

є псевдонормалізованою квазі- $\Phi$ -функцією, де  $\Phi'(u_1, u_2, u_p)$  визначено в (1.4).

## 1.6. Загальна задача завантаження контейнерів

На основі системного аналізу проблеми завантаження контейнерів та її місця в різних бізнес-процесах логістики і проектування можна зробити висновок щодо актуальності задач завантаження контейнерів та необхідності їх подальшого дослідження з метою підвищення ефективності розв'язання задач зазначеного класу. За результатами аналізу отримано, що задача завантаження контейнерів є оптимізаційною, має різні аспекти, які визначаються сферами застосування, і може аналізуватися на різних рівнях.

Загальну задачу завантаження контейнерів можна сформулювати у вигляді дворівневої задачі оптимізації так.

На першому (нижньому) рівні: завантажити (упакувати) заданий набір об'єктів, що мають різні просторові форми, метричні і механічні характеристики, в контейнер заданої форми і розмірів, за дотримання технологічних обмежень так, щоб критерій (вектор критеріїв) ефективності завантаження досягав екстремального значення.

На другому (верхньому) рівні: побудувати розклад обслуговування транспортних засобів на контейнерному терміналі і виконати переміщення заданого набору контейнерів у межах контейнерного терміналу так, щоб загальна вартість виконаних операцій була мінімальною за дотримання технологічних обмежень.

Як математичну модель досліджених оптимізаційних задач завантаження контейнерів використовують модель вигляду

$$\underset{w \in W}{extr} k(w), \text{ s.t. } w = (u, v) \in W,$$

$$W = \{w = (u, v) \in \mathbf{R}^\sigma : F(w) \geq 0\},$$

де  $W$  – множина допустимих розв'язків,  $k$  – цільова функція – відображення, задане на множині  $W$ ;  $w = (u, v)$  – вектор змінних моделі, серед яких  $u$  – неперервні,  $v$  – дискретні;  $F$  – система геометричних та технологічних обмежень.

## РОЗДІЛ 2. ОПТИМІЗОВАНЕ ПАКУВАННЯ КЛАСТЕРІВ ОБ'ЄКТІВ У ПРЯМОКУТНОМУ КОНТЕЙНЕРІ

Досліджується проблема пакування (компонування) для набору кластерів, що складаються з опуклих тривимірних об'єктів, основами яких є опуклі двовимірні фігури (наприклад, кола, еліпси або опуклі багатокутники). Ця проблема зводиться до двовимірної задачі розміщення. Кластери об'єктів мають бути упакованими в заданий прямокутний контейнер, без перетину між об'єктами в кожному кластері. Кожний кластер об'єктів наведено у вигляді опуклої оболонки об'єктів, що утворюють кластер. Два кластери об'єктів не перетинаються, якщо не перетинаються їхні опуклі оболонки. Вважають, що кластер повністю розміщений у контейнері, якщо це справедливо для його опуклої оболонки. Усі об'єкти в кластері мають однакову форму (допускаються різні розміри) і їх можна неперервно переміщувати та обертати. Задачею оптимізації пакування є побудова максимально розрідженого компонентування для кластерів об'єктів з урахуванням обмежень їх неперетину та включення у контейнер. Тут термін «розріджений» означає, що кластери досить віддалені один від одного. Наведено нові квазі- $\phi$ -функції та  $\phi$ -функції для аналітичного опису обмежень неперетину, включення у контейнер та обмежень на відстані між кластерами об'єктів. Задачу компонентування сформульовано як нелінійну неопуклу неперервну задачу оптимізації. Розроблено новий алгоритм пошуку локально-оптимальних розв'язків. Для підтвердження ефективності запропонованого підходу наведено результати обчислень. Це дослідження мотивовано проблемою завантаження контейнера, але подібні задачі виникають і в інших галузях науки і техніки.

### 2.1. Кластеризація пари нерегулярних об'єктів

Задачі оптимального розміщення (розкрою та пакування) цікаві як з теоретичної, так і з практичної точки зору. Прикладами застосування задач

розміщення у виробництві є розкрій металу, текстилю, скла, шкіри та фурнітури, 3D друк. Добре відомо, що навіть одновимірний випадок проблеми знаходження оптимального використання заданого ресурсу, класична задача про укладення ранцю, належить до класу NP-складних задач оптимізації. З огляду на це у більшості випадків, пов'язаних із задачами розміщення, застосовують евристичні підходи. Тим не менш, розроблення аналітичних методів математичного моделювання є актуальною проблемою для розширення класу задач розміщення, що можна розв'язати оптимально за допомогою сучасних NLP-solvers.

У дослідженні розглянуто задачу оптимального розміщення нерегулярних двовірних об'єктів в обмежувальній опуклій багатокутній зоні.

Побудова опуклої оболонки для заданої кількості фіксованих об'єктів – класична задача обчислювальної геометрії [209–211], що має важливі застосування в задачах пакування і розкрою, які виникають у виробництві, дослідженні операцій, механіці, хімії [212, 213]. Дослідниками запропоновано різні ефективні методи розв'язання цієї задачі (див., наприклад, [214, 215]). Задача ускладнюється, коли об'єкти вільно переміщуються і обертаються без взаємного перетину. У цьому випадку можуть бути побудовані різні опуклі оболонки залежно від параметрів розміщення об'єктів. Відповідно виникає задача побудови опуклої оболонки з мінімальним периметром (площею) [214, 216]. Цю задачу можна також інтерпретувати як пакування довільних об'єктів у мінімальний опуклий контейнер [217].

Поліноміальні методи розв'язання для побудови опуклих оболонок мініимального периметра відомі для двох або трьох багатокутників [218] і / або для багатокутників з обмеженням на трансляції / обертання [219].

У [220] розглянуто методи побудови опуклої оболонки довільного числа дисків з мінімальним периметром з аналітичним розв'язком, отриманими для трьох дисків. У [221] наведено чисельне та аналітичне розв'язання задачі про мінімальну поверхню опуклої оболонки сфер.

Розглянемо задачу кластерізації у такій постановці. Дано два довільних геометричних об'єкти, що обмежені дугами кіл та відрізками прямих. Над об'єктами можуть здійснюватися неперервні повороти та трансляції за умови їх неперетину. Знайти мінімальний розмір обмежувальної зони, що являє собою опуклу багатокутну оболонку мінімальної площі або периметра, та відповідні параметри розміщення об'єктів.

Між об'єктами та між кожним об'єктом і межею обмежувальної зони можуть бути задані мінімально допустимі відстані.

Припустимо, що будь-який розміщуваний об'єкт  $T$  (об'єкт, який має бути розміщений у визначеній зоні) — це двовимірний однозв'язний  $\phi$ -об'єкт.

*Визначення 2.1* [222]. Канонічно замкнута точкова множина  $T \subset R^2$  ( $T = cl^*(T) = cl(int(T))$ ), що має гомотопічний тип такий самий, як і гомотопічний тип її внутрішності ( $int(T)$ ).

Межа кожного однозв'язного  $\phi$ -об'єкта  $T$  задається послідовністю межових елементів  $l_1, l_2, \dots, l_n$  (нумерація ведеться за годинниковою стрілкою). Кожний елемент  $l_i$  задається вектором  $(x_i, y_i, r_i, x_{c_i}, y_{c_i})$ , якщо  $l_i$  є дугою або вектором  $(x_i, y_i, r_i)$ , якщо  $l_i$  є відрізком прямої, де  $(x_i, y_i)$  і  $(x_{i+1}, y_{i+1})$  — кінцеві точки  $l_i$ , а  $(x_{c_i}, y_{c_i})$  — центральна точка дуги. Кожен  $l_i$  є відрізком, якщо  $r_i = 0$ , “опуклою” дугою, якщо  $r_i > 0$ , “увігнутою” дугою, якщо  $r_i < 0$  (мається на увазі, що  $(x_{i+1}, y_{i+1}) = (x_1, y_1)$  для  $i = n$ ). Будемо називати компоненти векторів, що описують елементи  $l_1, l_2, \dots, l_n$ , метричними характеристиками розміщеного об'єкта  $T$ .

До такого упорядкованого набору відрізків прямих та дуг кіл застосовуємо алгоритм декомпозиції, наведений у [223], щоб отримати набір базових об'єктів, що описують складний об'єкт.

Кожний  $\phi$ -об'єкт  $T$ , обмежений відрізками прямих та дугами кіл, може бути декомпозований на  $n$  базових об'єктів чотирьох типів [223]. Позначимо

клас базових об'єктів як  $\mathfrak{R}$ . Кожен базовий об'єкт можна зобразити як перетин декількох примітивів (півплощина, круг, доповнення до круга).

Отже, подаємо розміщуваний об'єкт  $T$  у такому вигляді:

$$T = \bigcup_{j=1}^n T_j, \text{ де } T_j \in \mathfrak{R}. \quad (2.1)$$

Кожен розміщуваний об'єкт може бути описаний в аналітичній формі. З (2.1) отримуємо, що об'єкт  $T$  є об'єднанням базових об'єктів  $T_j$ ,  $j = 1, \dots, n$ .

Далі, кожен базовий об'єкт  $T_j$  є перетином примітивів  $T_{jk}$ ,  $j = 1, \dots, n$ . Тоді

$$T = \bigcup_{j=1}^n \bigcap_{k=1}^{n_j} T_{kj}. \text{ Визначимо } T_j = \{t \in R^2 : f_j(t) \leq 0\}, \text{ де } f_j(t) = \max_{k=1, \dots, n_j} f_{jk}(t),$$

$j = 1, \dots, n$ ,  $f_{jk}(t)$  – диференційована функція, пов'язана із примітивом  $T_{jk}$ .

Шукаємо максимум  $f_{jk}(t)$ ,  $k = 1, \dots, n_j$ , тому що об'єкт  $T_j$  визначається перетином примітивів. Отже, якщо  $\max_{k=1, \dots, n_j} f_{jk}(t) \leq 0$ , тоді  $f_{jk}(t) \leq 0$  для всіх

$k = 1, \dots, n_j$ . Оскільки  $T$  є об'єднанням простих об'єктів  $T_j$ , то шукаємо

$\min_{j=1, \dots, n} f_j(t)$ . Тож кожний розміщуваний об'єкт може бути поданий у вигляді

$$T = \{t \in R^2 : \min_{j=1, \dots, n} f_j(t) \leq 0\} \text{ або } T = \{t \in R^2 : \min_{j=1, \dots, n} \max_{k=1, \dots, n_j} f_{jk} \leq 0\}.$$

Місце знаходження та орієнтація у просторі розміщеного об'єкта  $T$  визначаються вектором його змінних параметрів  $(x_T, y_T, \theta_T)$ . Трансляцію

об'єкта  $T$  за вектором  $v_T = (x_T, y_T) \in R^2$  і поворот (відносно початку системи координат об'єкта) на кут  $\theta_T \in [0, 2\pi)$  визначають як

$T(x_T, y_T, \theta_T) = \{t \in R^2 : t = v_T + M(\theta_T)\tilde{t}, \forall \tilde{t} \in T(0, 0, 0)\}$ , де  $T(0, 0, 0)$  позначає об'єкт  $T$ , з яким не здійснювалося трансляцій та поворотів, а  $M(\theta_T)$  —

$$\text{стандартна матриця повороту } M(\theta_T) = \begin{pmatrix} \cos \theta_T & \sin \theta_T \\ -\sin \theta_T & \cos \theta_T \end{pmatrix}.$$

Уважаємо, що розміщені об'єкти мають фіксовані метричні

характеристики та змінні параметри розміщення  $(x_T, y_T, \theta_T)$ .

Обмежувальна зона  $\Omega$  є опуклим багатокутником, де кожна сторона  $e_i = [v_i, v_{i+1}]$ ,  $i = 1, \dots, m$  має змінну довжину  $t_i$  та визначається двома змінними вершинами  $v_i = (x_i, y_i)$  і  $v_{i+1} = (x_{i+1}, y_{i+1})$ ,  $x_{i+1} = x_i + t_i \cdot \cos \theta_i$ ,  $y_{i+1} = y_i + t_i \cdot \sin \theta_i$ . Тобто кожна сторона  $e_i$  може бути задана змінним вектором  $(x_i, y_i, \theta_i, t_i)$ .

Обмежувальна зона  $\Omega$  має фіксовані параметри розміщення і змінні метричні характеристики  $p$ , описані вище. У подальшому позначатимемо обмежувальну багатокутну зону  $\Omega = \Omega(p)$ .

*Аналitичний опис взаємодії між об'єктами.* Однією з успішних ідей у випадку математичного моделювання умов неперетину та включення для об'єктів складної форми, що дає змогу виконувати неперервні обертання та трансляції, є метод  $\phi$ -функцій [222]. Використання цього методу перетворює оптимізаційну задачу розміщення до багатоекстремальної задачі нелінійної оптимізації.

Для того, щоб знайти допустиме розміщення об'єктів усередині зони  $\Omega$ , необхідно мати деяке аналітичне подання взаємодії між парою об'єктів  $A$  і  $B$ . Для цього використаємо метод  $\phi$ -функцій.  $\phi$ -функції дають змогу розрізнити такі три випадки:  $A$  і  $B$  перетинаються, тобто мають спільні внутрішні точки,  $A$  і  $B$  не перетинаються, тобто не мають спільних точок,  $A$  і  $B$  дотинаються один до одного, тобто мають спільні точки тільки на межі.

*Визначення 2.2* [222].  $\phi$ -функцією двох об'єктів  $A$  і  $B$  називається всюди визначена неперервна функція, яка відповідає таким вимогам:

$$\text{а) } \Phi_{AB} > 0, \text{ якщо } A \cap B = \emptyset, \text{ б) } \Phi_{AB} = 0 \text{ якщо } \text{int } A \cap \text{int } B = \emptyset \text{ і}$$

$$\text{fr}A \cap \text{fr}B \neq \emptyset,$$

$$\text{в) } \Phi_{AB} < 0, \text{ якщо } \text{int } A \cap \text{int } B \neq \emptyset,$$

де  $\text{int } A$ ,  $\text{fr}A$  — внутрішність і межа об'єкта  $A$  відповідно.

Використаємо  $\phi$ -функції для опису відношення  $A \subseteq B$  належності

одного об'єкта іншому:  $\Phi_{AB^*} \geq 0$ , де  $B^* = R^2 \setminus \text{int } B$ . У працях [222, 223] наведено більш детальний опис властивостей  $\Phi$ -функцій.

Згідно з формулою (2.1) нехай

$$A = \bigcup_{i=1}^{n_A} A_i, \quad B = \bigcup_{j=1}^{n_B} B_j \quad \text{і} \quad A_i, B_j \in \mathfrak{R}. \quad (2.2)$$

$\Phi$ -функція, яка описує умову неперетину пари об'єктів  $A$  і  $B$  довільної форми, має вигляд

$$\Phi_{AB} = \min\{\Phi_{ij}, i = 1, 2, \dots, n_A, j = 1, 2, \dots, n_B\}, \quad (2.3)$$

де  $\Phi_{ij}$  позначає  $\Phi$ -функцію для пари базових об'єктів  $A_i, B_j \in \mathfrak{R}$  (або базову  $\Phi$ -функцію). Повний список базових  $\Phi$ -функцій наведено в [224].

Отже, у термінах  $\Phi$ -функцій можемо сформулювати умови включення (належності об'єкта контейнеру):

об'єкт  $A$  належить контейнеру  $\Omega$ , якщо  $\Phi_{\Omega^* A} \geq 0$ , де  $\Omega^* = R^2 \setminus \text{int } \Omega$ ,

$\Omega$  є контейнером, що має форму опуклого багатокутника.

Можна враховувати обмеження на мінімальні відстані, замінивши  $\Phi$ -функції на псевдонормалізовані  $\Phi$ -функції.

*Визначення 2.3.* Псевдонормалізованою  $\Phi$ -функцією об'єктів  $A$  і  $B$  називають всюди визначену неперервну функцію  $\hat{\Phi}^{AB}$ , яка відповідає таким вимогам:

а)  $\hat{\Phi}^{AB} > 0$ , якщо  $\text{dist}(A, B) > \rho$ ,

б)  $\hat{\Phi}^{AB} = 0$ , якщо  $\text{dist}(A, B) = \rho$ ,

в)  $\hat{\Phi}^{AB} < 0$ , якщо  $\text{dist}(A, B) < \rho$ ,

де  $\rho$  – мінімальна допустима відстань між об'єктами  $A$  та  $B$ .

Тут мається на увазі, що  $\text{dist}(A, B) = \min_{a \in A, b \in B} d(a, b)$  і  $d(a, b)$  — евклідова

відстань між двома точками  $a$  і  $b$  у  $R^2$ . Зокрема,  $\text{dist}(A, B) \geq \rho \Leftrightarrow \hat{\Phi}^{AB} \geq 0$ .

**Математична модель.** Сформулюємо задачу оптимального розміщення (або кластеризації) пари складених об'єктів в опуклій багатогранній зоні мінімального розміру у вигляді такої задачі умовної оптимізації:

$$\kappa(u^*) = \min \kappa(u), \quad \text{s.t. } u \in W \quad (2.4)$$

$$W = \{u \in R^\sigma : \widehat{\Phi}^{\Omega^* A} \geq 0, \widehat{\Phi}^{\Omega^* B} \geq 0, \widehat{\Phi}^{AB} \geq 0, \omega \geq 0\}, \quad (2.5)$$

де  $\kappa(u)$  – функція цілі (площа або периметр опуклої оболонки),  $u = (p, u_A, u_B) \in R^\sigma$  – вектор змінних,  $R^\sigma$  –  $\sigma$ -вимірний евклідовий простір,  $p$  – вектор змінних параметрів багатокутника  $\Omega$ ,  $(u_A, u_B) = (x_A, y_A, \theta_A, x_B, y_B, \theta_B)$  – вектор змінних параметрів розміщення об'єктів  $A$  і  $B$ ,  $W$  – множина допустимих розв'язків (простір розв'язків),  $\widehat{\Phi}^{AB}$  – псевдонормалізована  $\rho$ -функція для об'єктів  $A$  і  $B$ , яка враховує мінімальну допустиму відстань між об'єктами,  $\widehat{\Phi}^{\Omega^* A}$ ,  $\widehat{\Phi}^{\Omega^* B}$  – псевдонормалізовані функції для пар об'єктів  $A$  і  $\Omega^*$ ,  $B$  і  $\Omega^*$  відповідно, які враховують мінімальну допустиму відстань між кожним об'єктом та межею зони,  $\omega \geq 0$  – додаткова система обмежень на параметри  $p$ .

Зазначимо, якщо  $\rho = 0$ , то в (2.5) використовуємо звичайні  $\rho$ -функції замість псевдонормалізованих.

Розглядаємо два варіанти постановки задачі (2.4), (2.5) залежно від вигляду цільової функції  $\kappa(u)$ :

$$\text{Задача 1. } \kappa_1(u) = \sum_{i=1}^m t_i \quad (\text{периметр багатокутника } \Omega):$$

$$u = (x_1, y_1, \theta_1, t_1, \dots, x_m, y_m, \theta_m, t_m, u_A, u_B) \in R^\sigma,$$

$\sigma = 4m + 6$  ( $A$  і  $B$  можуть обертатись),  $\sigma = 4m + 4$  ( $A$  і  $B$  не можуть обертатись),  $\omega \geq 0$ .

$$\text{Тут } \{x_{i+1} - x_i - t_i \cdot \cos \theta_i = 0, y_{i+1} - y_i - t_i \cdot \sin \theta_i = 0,$$

$$(x_i - x_{i+1}) \cdot \cos \theta_i + (y_i - y_{i+1}) \cdot \sin \theta_i \geq 0, \text{ s.t. } m+1 \equiv 1, t_i \geq 0, \sum_{i=1}^m t_i > 0.$$

Задача 2.  $\kappa_2(u) = 0.5 \sum_{i=1}^m (x_i y_{i+1} - y_i x_{i+1})$  s.t.  $m+1 \equiv 1$  (площа

багатокутника  $\Omega$ )

$$u = (x_1, y_1, \theta_1, t_1, \dots, x_m, y_m, \theta_m, t_m, u_A, u_B) \in R^\sigma,$$

$\sigma = 4m + 6$  ( $A$  і  $B$  можуть обертатись),  $\sigma = 4m + 4$  ( $A$  і  $B$  не можуть обертатись),  $\omega \geq 0$ . Тут  $\{x_{i+1} - x_i - t_i \cdot \cos \theta_i = 0, y_{i+1} - y_i - t_i \cdot \sin \theta_i = 0,$

$$(x_i - x_{i+1}) \cdot \cos \theta_i + (y_i - y_{i+1}) \cdot \sin \theta_i \geq 0, \text{ s.t. } m+1 \equiv 1, t_i \geq 0, \sum_{i=1}^m t_i > 0.$$

На змінні параметри можна накладати додаткові обмеження. Наприклад, щоб зробити константою якусь змінну із вектора  $u = (p, u_A, u_B)$  або задати допустимі межі повороту об'єктів  $\theta_A, \theta_B$ .

Зазначимо, що  $W \neq \emptyset$ , тому що  $\kappa$  обмежена знизу, тож задача (2.4), (2.5) має розв'язок. Цільова функція є лінійною ( $\kappa_1$ ) або квадратичною ( $\kappa_2$ ). У (2.5)  $\phi$ -функції побудовано суперпозицією  $\min$  та  $\max$  лінійних і/або нелінійних функцій, включаючи тригонометричні функції  $\sin$  та  $\cos$ . Система  $\omega \geq 0$  включає лінійні і/або нелінійні функції. Множина  $W$  допустимих розв'язків є не опуклою, що призводить до багатоекстремальності. Отже, задача оптимальної кластеризації (2.4), (2.5) є задачею негладкої оптимізації.

**Загальна стратегія розв'язання.** Задача (2.4), (2.5) належить до класу задач негладкої нелінійної оптимізації, її не можна розв'язати безпосередньо. Опишемо підхід до побудови дерева розв'язків, яке визначає допустиму зону у вигляді об'єднання множини підзон, і стратегію знаходження локальних та глобальних оптимумів на сукупності таких підобластей.

Оскільки  $W = \bigcup_{s=1}^{\eta} W_s$  [14 222], можемо звести розв'язання задачі

негладкої оптимізації (2.4), (2.5) до розв'язання послідовності таких задач для  $u = (p, u_A, u_B)$ :

$$\kappa(u^*) = \min\{\kappa(u^{s*}), s = 1, \dots, \eta\}, \quad (2.6)$$

де

$$\kappa(u^{s*}) = \min \kappa(u) \text{ s.t. } u \in W_s. \quad (2.7)$$

Глобальний оптимум  $\kappa(u^*)$  можна отримати за допомогою знаходження і порівняння точного розв'язку всіх підзадач (2.7).

Підзадачі (2.7), у загальному випадку, є нелінійними задачами математичного програмування, їх можна розв'язати стандартними методами локальної оптимізації (такими як метод внутрішньої точки, метод допустимих напрямків) або NLP-solvers (такими як IPOPT, Baron).

*Дерево розв'язків.* Розглянемо  $\phi$ -функції  $\Phi_{AB}$ ,  $\Phi_{A\Omega^*}$  і  $\Phi_{B\Omega^*}$ , які беруть участь у формуванні множини  $W$  задачі (2.4), (2.5). Тут  $A$  і  $B$  — складені

об'єкти, що задано як  $A = \bigcup_{t=1}^{n_A} A_t$ ,  $B = \bigcup_{l=1}^{n_B} B_l$ ,  $A_t, B_l \in \mathfrak{R}$ , відповідно до (2.2).

Нехай  $\Phi_k \in \{\Phi_{A_t B_l}, \Phi_{A_t \Omega^*}, \Phi_{B_l \Omega^*}\}$  — базова  $\phi$ -функція,  $k = 1, 2, \dots, n$ , де  $n = n_A \cdot n_B + n_A + n_B$ ,  $t = 1, 2, \dots, n_A$ ,  $l = 1, 2, \dots, n_B$ ,  $n_A \cdot n_B$  — кількість базових  $\phi$ -функцій  $\Phi_{A_t B_l}$  у  $\phi$ -функції  $\Phi_{AB}$ ,  $n_A + n_B$  — кількість базових  $\phi$ -функцій  $\Phi_{A_t \Omega^*}$  і  $\Phi_{B_l \Omega^*}$  у  $\phi$ -функціях  $\Phi_{A\Omega^*}$  і  $\Phi_{B\Omega^*}$ .

Як зазначено в [225], кожен  $\phi$ -функцію можна подати у вигляді

$$\Phi_k = \max_{i=1, \dots, \eta_k} f_i^k = \max_{i=1, \dots, \eta_k} \min_{j=1, \dots, J_i^k} f_{ij}^k,$$

де  $f_{ij}^k$  — диференційовані функції. Оскільки  $\min_{j=1, \dots, J_i^k} f_{ij}^k \geq 0$  еквівалентно до

$f_{ij}^k \geq 0$  для всіх  $j$ ,  $\max_{i=1, \dots, \eta_k} f_i^k \geq 0$  означає, що хоча б одна із нерівностей,

наприклад  $f_{i_0}^k \geq 0$ , має бути виконаною.

Для кожної нерівності  $\Phi_k \geq 0$  можемо побудувати дерево, яке називається базовим  $\rho$ -деревом і позначається  $\mathfrak{T}_k$ , а  $\eta_k$  відповідає кількості кінцевих вузлів базового  $\rho$ -дерева. Кожний кінцевий вузол дерева  $\mathfrak{T}_k$  відповідає системі нерівностей  $f_i^k \geq 0, i = 1, 2, \dots, \eta_k$ .

Дерево розв'язків  $\mathfrak{T}$  описує допустиму множину  $W$  задачі (2.4), (2.5), його будують так.

Кореню дерева відповідає система нерівностей  $\omega \geq 0$ .

На першому рівні  $\rho$ -дерева  $\mathfrak{T}$  містить  $\tau_1 = \eta_1$  вузлів, де  $\eta_1$  — кількість кінцевих вузлів базового  $\rho$ -дерева  $\mathfrak{T}_1$ , яке описує  $\Phi_1 \geq 0, \Phi_1 = \max_{i=1, \dots, \eta_1} f_i^1$ ,

$f_i^1 = \min_{j=1, \dots, J_i^1} f_{ij}^1$ . Тут кожному вузлу відповідає система нерівностей

$$\{\omega \geq 0, f_{i_1}^1 \geq 0\}.$$

На другому рівні  $\rho$ -дерева  $\mathfrak{T}$  до кожного вузла на першому рівні додаємо  $\eta_2$  вузлів базового  $\rho$ -дерева  $\mathfrak{T}_2$ , яке описує  $\Phi_2 \geq 0$ , де

$$\Phi_1 = \max_{i=1, \dots, \eta_1} f_i^1, f_i^1 = \min_{j=1, \dots, J_i^1} f_{ij}^1, \text{ тобто кількість вузлів на другому рівні } \mathfrak{T}$$

становить  $\tau_2 = \eta_1 \cdot \eta_2$ . Тут кожному вузлу відповідає система нерівностей вигляду  $\{\omega \geq 0, f_{i_1}^1 \geq 0, f_{i_2}^2 \geq 0\}$ .

На  $k$ -му рівні  $\rho$ -дерева  $\mathfrak{T}$  до кожного вузла на  $(k-1)$ -му рівні додаємо  $\eta_k$  вузлів базового  $\rho$ -дерева  $\mathfrak{T}_k$ , яке описує  $\Phi_k \geq 0$ , де  $\Phi_k = \max_{i=1, \dots, \eta_k} f_i^k$ ,

$$f_i^k = \min_{j=1, \dots, J_i^k} f_{ij}^k, \text{ тобто кількість вузлів на } k\text{-му рівні } \mathfrak{T} \text{ становить}$$

$\tau_k = \eta_1 \cdot \eta_2 \cdot \dots \cdot \eta_k$ . Тут кожному вузлу відповідає система нерівностей вигляду  $\{\omega \geq 0, f_{i_1}^1 \geq 0, f_{i_2}^2 \geq 0, \dots, f_{i_k}^k \geq 0\}$ .

Зазначимо, що  $\tau_n = \eta_1 \cdot \eta_2 \cdot \dots \cdot \eta_{n-1} \cdot \eta_n = \eta$ , де  $\eta$  — кількість вузлів дерева розв'язків  $\mathfrak{T}$ .

Подамо множину  $W$  допустимих розв'язків задачі (2.4), (2.5) як об'єднання множин  $W_s, s = 1, 2, \dots, \eta$ . Кожна  $W_s$  відповідає  $s$ -му кінцевому вузлу дерева  $\mathfrak{T}$  і, отже,  $W_s$  визначається системою нерівностей вигляду  $\{\omega \geq 0, f_{s_k}^k \geq 0, k = 1, \dots, n\}$ .

Збільшення кількості кінцевих вузлів дерева розв'язків показує, що ця задача оптимальної кластеризації є NP-складною.

**Алгоритми розв'язання.** Для розв'язання задач оптимальної кластеризації (2.4), (2.5) запропоновано два алгоритми. Перший полягає у послідовному пошуку локальних екстремумів на всіх допустимих множинах і дає в результаті глобальний екстремум за умови, що кожна підзадача (2.7) може бути розв'язана оптимально. Другий алгоритм є значно швидшим і здійснює пошук локально-оптимального екстремуму тільки з однієї “перспективної” стартової точки.

*Алгоритм 1.* Цей алгоритм базується на побудові дерева розв'язків, описаній вище, і генерує системи нерівностей, які описують множини  $W_s \subset W$  з (2.7), використовуючи дерево розв'язків  $\mathfrak{T}$ . Алгоритм 1 застосовує прискорений пошук систем нерівностей, що відповідають вузлам  $v_{s_k}^k, s_k = 1, \dots, \eta_k, k = 1, \dots, n$  дерева  $\mathfrak{T}$ . Для того, щоб відкинути “неперспективні” вузли на  $k$ -му рівні дерева  $\mathfrak{T}$ , застосовуємо правила відтинання. Ці правила використовують несумісність отриманих систем нерівностей із верхніми обмеженнями цільової функції.

На останньому рівні дерева  $\mathfrak{T}$  розв'язуємо задачу оптимізації, що відповідає вузлу  $v_{s_n}^n$ :

$$\kappa(u_{s_n}^{n*}) = \min\{\kappa(u) : u \in W_s = V_{s_n}^n\},$$

$$V_{s_n}^n = \{u \in R^\sigma : f_{s_1}^1 \geq 0, f_{s_2}^2 \geq 0, \dots, f_{s_n}^n \geq 0, \omega \geq 0\}.$$

Якщо  $V_{s_n}^n \neq \emptyset$  і  $\kappa(u_{s_n}^{n*}) < \kappa^0$ , тоді задаємо  $\kappa^0 = \kappa(u_{s_n}^{n*})$  і приймаємо точку  $u_{s_n}^{n*}$  як найкращий розв'язок.

Складність алгоритму 1 залежить від числа  $\widehat{\eta} \leq \eta$  нелінійних задач оптимізації, які повинні бути розв'язані, і складності обраного методу нелінійної оптимізації ( $O(\eta)$ ). Алгоритм 1 може бути застосовано до “простих” об'єктів, оскільки кількість кінцевих вузлів дерева  $\mathfrak{Z}$  зростає дуже швидко із кількістю  $n_A$  і  $n_B$  базових об'єктів, з яких складаються  $A$  і  $B$ .

*Алгоритм 2.* Цей алгоритм знаходить хороший розв'язок, маючи при цьому прийнятний час обчислень, який не зростає суттєво із збільшенням складності об'єктів. Для того, щоб отримати хороший допустимий розв'язок  $u^0 \in W$ , алгоритм використовує швидкий евристичний метод, розглянутий у [222, 225]. Далі алгоритм застосовує ІРОРТ [226] для знаходження локального мінімуму задачі для кожної стартової точки. Опис цього алгоритму наведено нижче.

Визначимо функцію  $\Lambda(u) = \min\{\Phi_{AB}, \Phi_{\Omega^*A}, \Phi_{\Omega^*B}, \omega\}$ . Ціль алгоритму — виділити із  $\Lambda(u^0) \geq 0$  систему нерівностей, яка описує підмножину  $W_s \subset W$  таку, що  $u^0 \in W_s$ .

Формуємо підмножину  $W_s$  так. Виконуємо пошук вузлів  $v_s^1, s = 1, \dots, \eta_1$  першого рівня  $\mathfrak{Z}$  послідовно і шукаємо номер  $s_1$  такий, щоб  $f_{s_1}^1(u^0) = f^1(u^0) = \max\{f_1^1(u^0), f_2^1(u^0), \dots, f_{\eta_1}^1(u^0)\}$ . Потім виконуємо пошук дочірніх вузлів  $v_s^2, s = 1, \dots, \eta_2$  вузла  $v_{s_1}^1$  послідовно і шукаємо номер  $s_2$  такий, щоб  $f_{s_2}^2(u^0) = f^2(u^0) = \max\{f_1^2(u^0), f_2^2(u^0), \dots, f_{\eta_1}^2(u^0)\}$ .

На  $n$ -му рівні дерева розв'язків  $\mathfrak{Z}$  виконуємо пошук дочірніх вузлів  $v_s^n, s = 1, \dots, \eta_n$  вузла  $v_{s_{n-1}}^{n-1}$  послідовно і шукаємо номер  $s_n$  такий, щоб  $f_{s_n}^n(u^0) = f^n(u^0) = \max\{f_1^n(u^0), f_2^n(u^0), \dots, f_{\eta_n}^n(u^0)\}$ . Тоді формуємо систему нерівностей, яка відповідає  $s$ -му кінцевому вузлу дерева  $\mathfrak{Z}$  у вигляді  $W_s = \{u \in R^\sigma : f_{s_1}^1 \geq 0, f_{s_2}^2 \geq 0, \dots, f_{s_n}^n \geq 0, \omega \geq 0\}$ . Кожній послідовності чисел

$s_1, s_2, \dots, s_k, \dots, s_n$  відповідає число  $s$ , яке описує формула

$$s = s_1^* = \begin{cases} s_k^* = (s_k - 1) \cdot \prod_{i=k+1}^n \eta_i + s_{k+1}^*, \text{ if } k = 1, \dots, n-1 \\ s_k^* = s_k, \text{ if } k = n \end{cases}.$$

Далі розв'язуємо задачу  $\min_{u \in W_s} \kappa(u)$ , стартуючи із точки  $u^0$ .

Складність алгоритму 2 залежить від кількості  $n$  базових  $\phi$ -функцій, що складають простір розв'язків, і складності обраного методу нелінійної оптимізації ( $O(n_A \cdot n_B)$ ).

Отже, запропонований підхід дає змогу знайти глобальний мінімум задачі (2.4), (2.5) для складених об'єктів, використовуючи алгоритм 1, оскільки у цьому випадку кожна задача (2.7) перетворюється на лінійну задачу, а також знайти хороше наближення глобального мінімуму у задачах кластеризації, використовуючи алгоритм 1 або алгоритм 2.

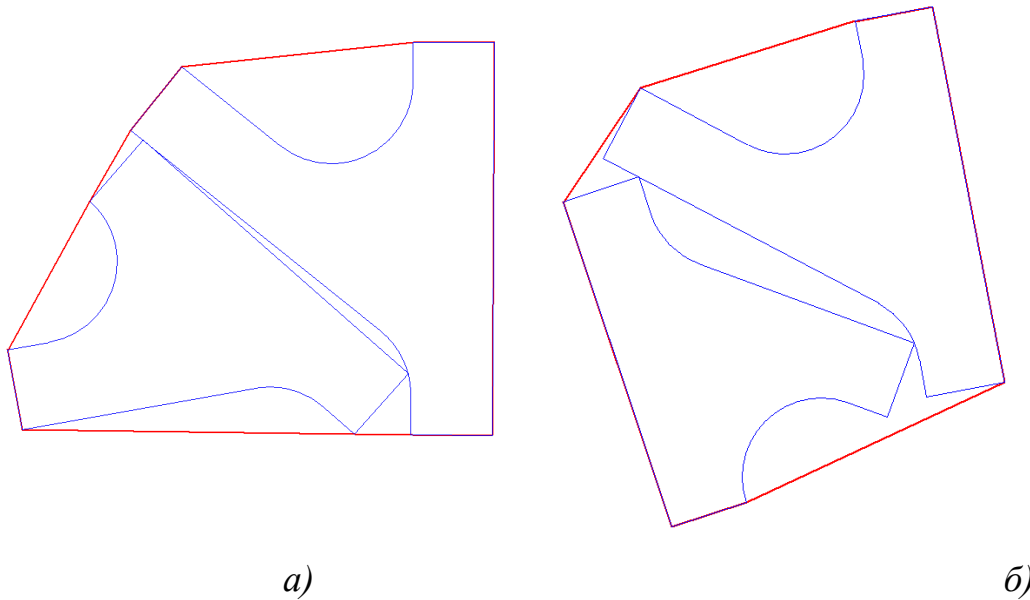
**Обчислювальні експерименти.** Наведено приклади розв'язання задачі 1 та задачі 2 за допомогою запропонованого на базі теоретичних розроблень програмного комплексу побудови опуклих багатокутних оболонки мінімальної площі і периметра для пари довільних об'єктів.

*Приклад 2.1.* Об'єкти задано у спосіб, описаний у розділі 1, у вигляді

об'єкт А: {115.182702089, -204.598967565, 0.0, 286.156172033, -33.625497622, 0.0, 215.484384787, 37.037804343, 0.0, 179.988331478, 1.488010919, -100.048312734, 109.256014172, 72.245372157, 38.511173909, 1.500531658, -100.048313476, 109.256014172, 72.245372157, 9.843590682, 60.983268381, 0.0, -7.520830543, 214.904151189, 0.0, -106.889839396, 203.694387387, 0.0, -62.729606698, -187.750027183, -99.944086869, -162.035487006, -199.026668367, -91.325712036, -269.659155395, 0.0, -126.568621117, -304.839839080, 0.0, -55.790767854, -375.572437508, 0.0},

об'єкт В: {96.204615870, -171.078974587, 0.0, 249.486981255, -17.796609202, 0.0, 178.750140186, 53.025084682, 0.0, 178.656802090, 52.975587207, -99.991451813, 107.927676864, 123.655720769, 178.631998906,

52.950775537, -99.991321995, 107.927676864, 123.655720769, 37.222915214,  
 52.950775537, -99.991322629, 107.927676864, 123.655720769, 8.572165371,  
 112.395184244, 0.0, 2.694693806, 164.217626023, 0.0, -96.668658193,  
 152.948465251, 0.0, -63.853246692, -136.377002193, -99.939811127,  
 -163.147698454, -147.716212810, -92.399854555, -218.304515182, 0.0,  
 -127.796912915, -253.698745114, 0.0, -57.077042408, -324.360632865, 0.0}.



**Рис. 2.1.** Мінімальна багатогранна опукла оболонка: *а* – мінімальної площі, *б* – мінімального периметра

Локально-оптимальне розміщення складених об'єктів  $A$  і  $B$  для прикладу 2.1 наведено на рис. 2.1. Мінімальний периметр опуклої багатокутної оболонки, отриманий за 0,717 с, дорівнює 1935,287432. Мінімальна площа опуклої багатокутної оболонки, отриманої за 1.046 с, дорівнює 480903,019848.

*Приклад 2.2* Розглядаємо два об'єкти, описані за допомогою списків:

об'єкт  $A$ : {51.468684320, 7.214667833, -5.309998523, 55.424284158,  
 3.672166705, 60.218684320, 5.954667833, 11.249999735, 65.284886527,  
 15.999372370, 72.398684320, 24.714667833, 62.809988100, 31.355167592,  
 -22.830303927, 0.528684320, 31.894667833, 6.560005046, 3.958629106,  
 26.302785773, -2.601315680, 26.274667833, 0.0, -2.601315680, 19.714667833,

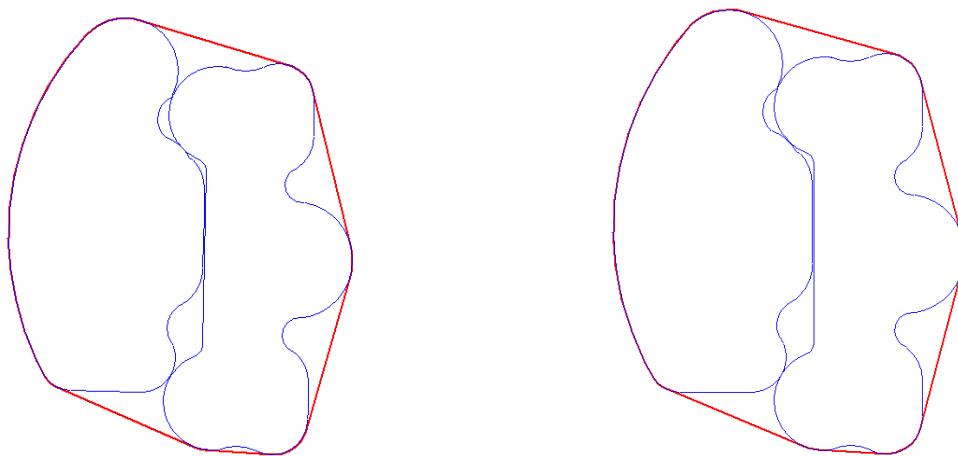
0.0, -2.601315680, 12.214667833, 7.288785186, 4.648795476, 11.464816041,  
 7.398684320, 4.714667833, -5.620000983, 11.164850519, 0.543288779,  
 15.528684320, 4.084667833, 8.749994118, 23.582012992, 7.506114072,  
 23.968684320, -1.235332167, 0.0, 35.218684320, -1.235332167, 0.0,  
 40.218684320, -1.235332167, 9.380010190, 40.390479062, 8.143104682,  
 46.778684320, 1.274667833, -5.630010326, 51.764902963, -1.339646591,  
 47.398684320,

2.214667833, 9.379987951, 42.602157727, 10.275530480},

об'єкт В: {38.778048354, 5.574803081, 12.190011326,  
 38.922193441, 17.763962129, 50.968048354, 15.894803081, -3.749999419,  
 54.692439191, 16.332305691, 58.158048354, 17.764803081, 8.119995261,  
 64.789337013, 22.450988496, 65.038048354, 14.334803081, 0.0, 70.658048354,  
 14.334803081, 0.0, 72.538048354, 14.334803081, 7.499997814, 72.739732315,  
 21.832088642, 79.408048354, 25.264803081, -9.690008705, 88.274173139,  
 29.174803081, 79.408048354, 33.084803081, 10.310000546, 69.229602639,  
 34.726556440, 59.718048354, 38.704803081, -2.500001698, 57.298393041,  
 39.333513392, 57.218048354, 36.834803081, 0.0, 38.778048354, 36.834803081,  
 0.0, 20.348048354, 36.834803081, -2.500001698, 20.267703667, 39.333513392,  
 17.848048354, 38.704803081, 10.310000546, 8.336494068, 34.726556440,  
 -1.841951646, 33.084803081, -9.690008705, -10.708076431, 29.174803081,  
 -1.841951646, 25.264803081, 7.499999549, 4.826223262, 21.831810708,  
 5.038048354, 14.334803081, 0.0, 6.908048354, 14.334803081, 0.0, 12.538048354,  
 14.334803081, 8.120009804, 12.776960205, 22.451297419, 19.408048354,  
 17.764803081, -3.750000599, 22.873655350, 16.332297359, 26.598048354,  
 15.894803081, 12.189988489, 38.643866050, 17.764053036}.

Локально-оптимальне розміщення складених об'єктів  $A$  і  $B$  для прикладу 2.2 наведено на рис. 2.2. Мінімальна площа опуклої багатокутної оболонки, отриманої за 11,528 с, дорівнює 10479,396755.

Мінімальний периметр опуклої багатокутної оболонки, отриманий за 6,068 с, дорівнює 266,648192.



**Рис. 2.2.** Мінімальна багатогранна опукла оболонка: *a* – мінімальної площі, *б* – мінімального периметра

## 2.2. Задача компоновання кластерів об'єктів

Як відомо [117], задачі пакування об'єктів у контейнер є NP-складними. Задача компоновання кластерів об'єктів характерна для логістики (транспортування рулонів шпалер, труб, коробок, відер для фарби тощо), а також має важливе застосування в інформатиці, промисловому машинобудуванні, виробничих процесах, охороні здоров'я, виборі портфелю проєктів, нанofізичі, сільському господарстві (див., наприклад, [135] та посилання в ній).

У задачах пакування досліджували різні форми об'єктів. Зокрема, задачі розкрою та пакування для еліпсів розглянуто в [25, 136–138], задачі пакування кіл досліджено в [26, 139–145], задачі пакування для опуклих багатокутників розглянуто в [143–148]. Праці [21, 129, 149–151] присвячено нерегулярному пакуванню об'єктів довільної форми.

У багатьох випадках об'єкти не є незалежними, і їх доводиться об'єднувати в кластери об'єктів, що не перетинаються. Це характерно, наприклад, для проблеми завантаження контейнера [152], коли об'єкти у великому морському контейнері можуть утворювати різні кластери відповідно до типу об'єктів (подібні фігури, деталі однієї машини). Аналогічно, кластери можуть бути сформовані відповідно до постачальника або клієнта (кінцевого

пункту призначення) для полегшення завантаження або розвантаження контейнера.

Хоча кластери (кількість об'єктів та їх форми), зазвичай, заздалегідь визначено, загальну форму кластерів об'єктів часто не вказано. Розглядаючи кластер як набір об'єктів, поміщених у гнучкий «мішок», визначаємо форму кластера як опуклу оболонку множини об'єктів у кластері. Зазначимо, що об'єкти не перетинаються, і форма кластера (опукла оболонка) залежить від розташування об'єктів у кластері.

У цьому розділі передбачено однорідність кластерів, що складаються з однакових форм (допускаються різні розміри). Задано кількість кластерів, а також кількість об'єктів та їх форми та розміри. Форма кластера зображена як опукла оболонка об'єктів кластерів. Допустиме розташування кластерів має відповідати таким умовам: об'єкти в кластері взаємно не перетинаються; кластери об'єктів взаємно не перетинаються та не перетинаються з доповненням до прямокутного контейнера.

Для оптимізації пакування кластерів об'єктів можна використовувати різні цілі. Наприклад, можна шукати найщільніше компонування, за якої фіксується один розмір прямокутного контейнера і мінімізується інший розмір до можливості здійснення компонування кластерів об'єктів. Навпаки, можна зафіксувати обидва розміри контейнера і шукати „найрідкіснішу” схему, яка максимізує певну відстань між кластерами об'єктів. Ця ціль мотивована необхідністю більшого простору між кластерами для полегшення доступу для їх завантаження / розвантаження.

Сформулюємо постановку задачі.

Нехай  $\Omega = \{(x, y) \in R^2 : 0 \leq x \leq l, 0 \leq y \leq w\}$  – прямокутна зона (контейнер) і  $T_i, i \in \{1, 2, \dots, n\} = I_n$ , – упорядкований набір опуклих об'єктів. Кожен об'єкт  $T_i$  визначається його метричними характеристиками. Зокрема, еліпс  $E_i$  визначається його півсями  $a_i$  та  $b_i$ , коло  $C_i$  – його радіусом  $r_i$ , опуклий багатокутник  $K_i$  – його вершинами  $p_{ik} = (p_{ik}^x, p_{ik}^y), k = 1, \dots, l_i$ .

Положення об'єкта  $T_i$  характеризується вектором змінних параметрів розміщення  $u_i = (v_i, \theta_i)$ , де  $v_i = (x_i, y_i)$  – вектор трансляції,  $\theta_i$  – кут повороту. Центр об'єкта збігається з початком його локальної системи координат. Об'єкт  $T_i$ , повернений на кут  $\theta_i$  і трансльований на вектор  $v_i$ , визначається як  $T_i(u_i) = \{p \in R^2 : p = v_i + M(\theta_i) \cdot p^0, \forall p^0 \in T_i^0\}$ , де  $T_i^0$  позначає нетрансльований і неповернений об'єкт  $T_i$ ,

$$M(\theta_i) = \begin{pmatrix} \cos \theta_i & \sin \theta_i \\ -\sin \theta_i & \cos \theta_i \end{pmatrix} \text{ – стандартна матриця обертання.}$$

*Зауваження.* Положення кола  $C_i$  характеризується вектором руху  $u_i = (v_i, 0) = (x_i, y_i, 0)$ .

Визначаємо кластер об'єктів, що мають однакову форму. Впорядкований набір об'єктів  $T_i$ ,  $i = 1, \dots, n$ , розділений на  $N$  кластерів  $\Lambda_1 = \{T_1, \dots, T_{n_1}\}$ ,  $\Lambda_2 = \{T_{n_1+1}, \dots, T_{n_2}\}$ ,  $\dots$ ,  $\Lambda_N = \{T_{n_{N-1}+1}, \dots, T_n\}$  за допомогою наборів індексів  $\Xi_1 = \{1, \dots, n_1\}$ ,  $\Xi_2 = \{n_1 + 1, \dots, n_2\}$ ,  $\dots$ ,  $\Xi_N = \{n_{N-1} + 1, \dots, n\}$ ,  $n = \sum_{q=1}^N m_q$ , де  $m_q = \text{card}(\Xi_q)$ ,  $q \in \{1, 2, \dots, N\} = J_N$ .

Уведемо вектори параметрів розміщення кластерів  $\Lambda_1, \Lambda_2, \dots, \Lambda_N$  у вигляді  $z_1 = (u_1, \dots, u_{n_1})$ ,  $z_2 = (u_{n_1+1}, \dots, u_{n_2})$ ,  $\dots$ ,  $z_N = (u_{n_{N-1}+1}, \dots, u_n)$  і використаємо позначення  $\Lambda_q(z_q)$  для кластера  $\Lambda_q$ , що включає рухомі об'єкти  $T_i(u_i)$ ,  $i \in \Xi_q$ .

З кожним кластером  $\Lambda_q(z_q) = \bigcup_{i \in \Xi_q} T_i(u_i)$  зв'язуємо опуклу оболонку об'єктів  $T_i(u_i)$ ,  $i \in \Xi_q$ , що позначаємо як  $\widehat{\Lambda}_q(z_q) = \text{conv} \Lambda_q(z_q)$ ,  $q \in J_N$ .

У цьому розділі використовуємо такі визначення: 1) два кластери  $\Lambda_q(z_q)$  і  $\Lambda_g(z_g)$  не перетинаються, якщо їхні опуклі оболонки  $\widehat{\Lambda}_q(z_q)$  і

$\widehat{\Lambda}_g(z_g)$  не перетинаються,  $q > g \in J_N$ ; 2) кластер  $\Lambda_q(z_q)$  належить контейнеру  $\Omega$ , якщо його опукла оболонка  $\widehat{\Lambda}_q(z_q)$  належить контейнеру  $\Omega$ ,  $q \in I_N$ .

*Задача компоювання кластерів об'єктів (Group Layout Problem, GLP).* Розмістити набір кластерів  $\Lambda_q(z_q)$ ,  $q \in J_N$ , достатньо віддалено один від одного в прямокутному контейнері  $\Omega$  так, щоб

$$\text{int } T_i(u_i) \cap \text{int } T_j(u_j) = \emptyset \text{ для } i > j, (i, j) \in \Sigma_q \times \Sigma_q, q \in J_N, \quad (2.8)$$

$$\text{int } \widehat{\Lambda}_q(z_q) \cap \text{int } \widehat{\Lambda}_g(z_g) = \emptyset \text{ для } q > g \in J_N, \quad (2.9)$$

$$\widehat{\Lambda}_q(z_q) \subset \Omega \text{ для кожного } q \in I_N. \quad (2.10)$$

Перше обмеження забезпечує неперетин об'єктів однієї групи, друге гарантує неперетин кластерів, третє обмеження визначає умови належності кластерів контейнеру  $\Omega$ .

2.3. Засоби математичного моделювання компоювання кластерів об'єктів у прямокутному контейнері

Використано метод  $\phi$ -функцій (див. п. 1.5). Наведено нові засоби для опису обмежень розміщення (2.9), (2.10) в аналітичному вигляді.

*Неперетин об'єктів у кластері.* Для опису обмеження (2.8) використовуємо  $\phi$ -функції та квазі- $\phi$ -функції для пари різних форм опуклих об'єктів.

Розглянемо два об'єкти  $T_i(u_i)$  та  $T_j(u_j)$ , які належать до кластера  $\Lambda_q(z_q)$ ,  $i > j, (i, j) \in \Sigma_q \times \Sigma_q, q \in J_N$ .

Згідно з визначенням  $\phi$ -функції [122] маємо  $\Phi^{T_i T_j}(u_i, u_j) \geq 0$ , якщо і тільки якщо  $\text{int } T_i(u_i) \cap \text{int } T_j(u_j) = \emptyset$ .

З основної властивості квазі- $\phi$ -функції [25] випливає, що якщо  $\Phi^{T_i T_j}(u_i, u_j, u_{ij}) \geq 0$  для деякого  $u_{ij}$ , то  $\text{int } T_i(u_i) \cap \text{int } T_j(u_j) = \emptyset$ , де  $u_{ij}$  – вектор додаткових змінних.

Зокрема, пропонуємо засоби опису обмеження неперетину для пари еліпсів, кіл та опуклих багатокутників.

*Квазі- $\phi$ -функція для еліпсів.* Нехай  $E_i(u_i)$  та  $E_j(u_j)$  – еліпси, задані їх півосями  $(a_i, b_i)$  та  $(a_j, b_j)$ , з параметрами розміщення  $u_i = (x_i, y_i, \theta_i)$  і  $u_j = (x_j, y_j, \theta_j)$ .

Квазі-  $\phi$ -функція для  $E_i(u_i)$  та  $E_j(u_j)$  може бути визначена як

$$\Phi^{E_i E_j}(u_i, u_j, u_{ij} = \phi_{ij}) = x'_{ij}(v_i, \phi_{ij}) - x'_{ji}(v_j, \phi_{ij}) - (d_{ij}(\theta_i, \phi_{ij}) + d_{ji}(\theta_j, \phi_{ij})),$$

де

$$x'_{ij}(v_i, \phi_{ij}) = x_i \cos \phi_{ij} - y_i \sin \phi_{ij}, \quad x'_{ji}(v_j, \phi_{ij}) = x_j \cos \phi_{ij} - y_j \sin \phi_{ij},$$

$$d_{ij}(\theta_i, \phi_{ij}) = \sqrt{b_i^2 + (a_i^2 - b_i^2) \cos^2(\theta_i - \phi_{ij})},$$

$$d_{ji}(\theta_j, \phi_{ij}) = \sqrt{b_j^2 + (a_j^2 - b_j^2) \cos^2(\theta_j - \phi_{ij})},$$

$\phi_{ij} \in R^1$  – допоміжна змінна (кут між прямою, що проходить через початок глобальної системи координат  $XOY$ , та віссю  $OX$ ).

*$\phi$ -функція для кіл.* Нехай  $C_i(v_i)$  та  $C_j(v_j)$  – кола, задані своїми радіусами  $r_i$  та  $r_j$  з параметрами розміщення  $v_i = (x_i, y_i)$  та  $v_j = (x_j, y_j)$ .

$\phi$ -функція для  $C(v_i)$  та  $C_j(v_j)$  має вигляд [122]

$$\Phi^{C_i C_j}(v_i, v_j) = (x_i - x_j)^2 + (y_i - y_j)^2 - (r_i + r_j)^2.$$

*Квазі- $\rho$ -функція для опуклих багатокутників.* Нехай опуклі багатокутники  $K_i(u_i)$  та  $K_j(u_j)$  визначені їх вершинами  $p_{ik} = (p_{ik}^x, p_{ik}^y)$ ,  $k=1, \dots, l_i$ ,  $p_{jk} = (p_{jk}^x, p_{jk}^y)$ ,  $k=1, \dots, l_j$ , та параметрами розміщення  $u_i = (x_i, y_i, \theta_i)$  і  $u_j = (x_j, y_j, \theta_j)$  та нехай  $P(u_{ij}) = \{(x, y) : \mu_{ij} = \cos \phi_{ij} \cdot x + \sin \phi_{ij} \cdot y + \gamma_{ij} \leq 0\}$  – півплощина.

Згідно з [130] квазі- $\rho$ -функцію для  $K_i(u_i)$  та  $K_j(u_j)$  можна записати у формі

$$\Phi^{K_i K_j}(u_i, u_j, u_{ij}) = \min \{ \Phi^{K_i P}(u_i, u_{ij}), \Phi^{K_j P^*}(u_j, u_{ij}) \},$$

де  $u_{ij} = (\phi_{ij}, \gamma_{ij}) \in R^2$  є вектором допоміжних змінних,

$$\Phi^{K_i P}(u_i, u_{ij}) = \min_{1 \leq k \leq l_i} \mu_{ij}(p_{ik})$$

є  $\rho$ -функцією для  $K_i(u_i)$  і  $P(u_{ij})$  та

$$\Phi^{K_j P^*}(u_j, u_{ij}) = \min_{1 \leq k \leq l_j} (-\mu_{ij}(p_{jk}))$$

є  $\rho$ -функцією  $K_j(u_j)$  та  $P^*(u_{ij}) = R^2 \setminus \text{int } P(u_{ij})$ .

*Неперетин кластерів об'єктів.* Опишемо аналітично обмеження розміщення (2.9). З цією метою вводимо квазі- $\rho$ -функцію для двох конфігурацій  $\hat{\Lambda}_q(z_q)$  та  $\hat{\Lambda}_g(z_g)$ .

Розглянемо розширення методу  $\rho$ -функцій на опуклі оболонки кластерів об'єктів. Зазначимо, що опукла оболонка об'єктів у кластері має змінну форму та змінні метричні характеристики, які залежать від змінних параметрів розміщення об'єктів у кластері.

Нехай  $P_{qg} = \{(x, y) : \mu_{qg}(x, y) = \cos \varphi_{qg} \cdot x + \sin \varphi_{qg} \cdot y + \gamma_{qg} \geq 0\}$  – півплощина зі змінними параметрами розміщення  $(\varphi_{qg}, \gamma_{qg})$ .

*Твердження 2.1.* Неперервна і всюди визначена функція

$$\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) = \min\{\Phi^{\widehat{\Lambda}_q P_{qg}}(z_q, \varphi_{qg}, \gamma_{qg}), \Phi^{\widehat{\Lambda}_g P_{qg}^*}(z_g, \varphi_{qg}, \gamma_{qg})\} \quad (2.11)$$

є квазі- $\rho$ -функцією для опуклих оболонок  $\widehat{\Lambda}_q(z_q)$  та  $\widehat{\Lambda}_g(z_g)$ , де  $\Phi^{\widehat{\Lambda}_q P_{qg}}(z_q, \varphi_{qg}, \gamma_{qg})$  –  $\rho$ -функція для опуклої оболонки  $\widehat{\Lambda}_q(z_q)$  і півплощини  $P_{qg}$ ,  $\Phi^{\widehat{\Lambda}_g P_{qg}^*}(z_g, \varphi_{qg}, \gamma_{qg})$  –  $\rho$ -функція для опуклої оболонки  $\widehat{\Lambda}_g(z_g)$  і півплощини  $P_{qg}^* = R^2 \setminus \text{int } P_{qg}$ .

*Доведення.* На основі властивостей квазі- $\rho$ -функції [25], визначеної для двох опуклих об'єктів, можна зробити висновок, що  $\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) \geq 0$  тоді і тільки тоді, коли існує хоча б одна відокремлювальна лінія  $L_{qg}\{(x, y) : \mu_{qg}(x, y) = 0\}$  для деякого  $\varphi_{qg}, \gamma_{qg}$  така, що  $\widehat{\Lambda}_q(z_q) \subset P_{qg}$  та  $\widehat{\Lambda}_g(z_g) \subset P_{qg}^*$  і, отже,  $\widehat{\Lambda}_q(z_q) \cap \widehat{\Lambda}_g(z_g) = \emptyset$  для  $q > g \in J_N$ .

Визначимо  $\rho$ -функцію для опуклої оболонки  $\widehat{\Lambda}_q(z_q)$  і півплощини  $P_{qg}$  у (2.11) у вигляді

$$\Phi^{\widehat{\Lambda}_q P_{qg}}(z_q, \varphi_{qg}, \gamma_{qg}) = \min\{\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\}, \quad (2.12)$$

де  $\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg})$  є  $\rho$ -функцією об'єкта  $T_i(u_i)$  і півплощини  $P_{qg}$ ,  $i \in \Xi_q$ .

З властивостей опуклої оболонки  $\Lambda_q(z_q) = \bigcup_{i \in \Xi_q} T_i(u_i)$  та характеристик

відокремлювальної прямої між двома опуклими об'єктами можна зробити висновок, що

1)  $\widehat{\Lambda}_q(z_q) \cap P_{qg} = \emptyset \Leftrightarrow \Lambda_q(z_q) \cap P_{qg} = \emptyset$ , отже,  $T_i(u_i) \cap P_{qg} = \emptyset$ , для всіх  $i \in \Xi_q$ . З цього випливає, що  $\min\{\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\} > 0$ ;

2)  $\text{int } \widehat{\Lambda}_q(z_q) \cap \text{int } P_{qg} \neq \emptyset \Leftrightarrow \text{int } \Lambda_q(z_q) \cap \text{int } P_{qg} \neq \emptyset$ , отже, існує принаймні один об'єкт  $T_i(u_i)$  такий, що  $\text{int } T_i(u_i) \cap \text{int } P_{qg} \neq \emptyset$ ,  $i \in \Xi_q$ . З цього випливає, що  $\min\{\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\} < 0$ ;

3)  $(\text{int } \widehat{\Lambda}_q(z_q) \cap \text{int } P_{qg} = \emptyset) \wedge (\text{fr } \widehat{\Lambda}_q(z_q) \cap \text{fr } P_{qg} \neq \emptyset) \Leftrightarrow$   
 $(\text{int } \Lambda_q(z_q) \cap \text{int } P_{qg} = \emptyset) \wedge (\text{fr } \Lambda_q(z_q) \cap \text{fr } P_{qg} \neq \emptyset)$  і, отже,  
 $\text{int } T_i(u_i) \cap \text{int } P_{qg} = \emptyset$ , для всіх  $i \in \Xi_q$ . Тоді існує принаймні один об'єкт  $T_i(u_i)$  такий, що  $\text{fr } T_i(u_i) \cap \text{fr } P_{qg} \neq \emptyset$ ,  $i \in \Xi_q$ . З цього випливає, що  $\min\{\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}), i \in \Xi_q\} = 0$ .

Таким чином, функція  $\Phi^{\widehat{\Lambda}_q P_{qg}}(z_q, \varphi_{qg}, \gamma_{qg})$ , визначена у (2.12), є рхі-функцією для опуклої оболонки  $\widehat{\Lambda}_q$  і півплощини  $P_{qg}$ .

Так само рхі-функція для опуклої оболонки  $\widehat{\Lambda}_g(z_g)$  і півплощини  $P_{qg}^* = R^2 \setminus \text{int } P_{qg}$  набуває форми

$$\Phi^{\widehat{\Lambda}_g P_{qg}^*}(z_g, \varphi_{qg}, \gamma_{qg}) = \min\{\Phi^{T_j P_{qg}^*}(u_j, \varphi_{qg}, \gamma_{qg}), j \in \Xi_g\},$$

де  $\Phi^{T_j P_{qg}^*}(u_j, \varphi_{qg}, \gamma_{qg})$  є рхі-функцією об'єкта  $T_j(u_j)$ ,  $j \in \Xi_g$ , і півплощини  $P_{qg}^*$ .

Отже,  $\Phi^{\widehat{\Lambda}_q P_{qg}}(z_q, \varphi_{qg}, \gamma_{qg}) \geq 0$ , якщо і тільки якщо  $\Phi^{T_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}) \geq 0$  для кожного  $i \in \Xi_q$ .

Можна зробити висновок, що  $\max_{(\varphi_{qg}, \gamma_{qg})} \Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$  є  $\phi$ -функцією для опуклих оболонок  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$ , отже,  $\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$  є квазі- $\phi$ -функцією цих опуклих оболонок.

Зазначимо, що квазі- $\phi$ -функція  $\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) - 0,5\rho$  є нормалізованою квазі- $\phi$ -функцією для опуклих оболонок  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$  (див. п. 1.5). Це означає, що якщо  $\rho > 0$  – є відстань між  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$  визначається стандартним способом як мінімальна евклідова відстань між кожною парою точок, що належать відповідним опуклим оболонкам, тоді нерівність  $\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) - 0,5\rho \geq 0$  означає, що  $\text{dist}(\widehat{\Lambda}_q(z_q), \widehat{\Lambda}_g(z_g)) \geq \rho$ .

Зазначимо, що для опису обмежень неперетину (2.9) опуклі оболонки кластерів об'єктів не будують в явному вигляді завдяки особливостям квазі- $\phi$ -функцій для опуклих оболонок  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$ .

Далі розглянемо засоби аналітичного опису неперетину об'єктів –  $\phi$ -функції для півплощини

$$P_{qg} = \{(x, y) : \mu_{qg}(x, y) = \cos \varphi_{qg} \cdot x + \sin \varphi_{qg} \cdot y + \gamma_{qg} \geq 0\}$$

та таких форм об'єктів  $T_i(u_i)$ ,  $i \in \Xi_q$ : еліпса  $E_i(u_i)$ ; кола  $C_i(v_i)$ ; і опуклого багатокутника  $K_i(u_i)$ .

$\phi$ -функція для еліпса і півплощини  $P_{qg}$ .  $\phi$ -функцію для еліпса  $E_i(u_i)$ , заданого його півосями  $(a_i, b_i)$ , і півплощини  $P_{qg}$  можна визначити у формі

$$\Phi^{E_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}) = x_i \cos \varphi_{qg} + y_i \sin \varphi_{qg} + \gamma_{qg} - \sqrt{(a_i^2 - b_i^2) \cdot \cos^2(\theta_i + \varphi_{qg}) + b_i^2}.$$

*Phi-функція для кола і півплощини  $P_{qg}$ .* Phi-функцію для кола  $C_i(v_i)$ , заданого його радіусом  $r_i$ , і півплощини  $P_{qg}$  визначено в [87], вона має вигляд

$$\Phi^{C_i P_{qg}}(v_i, \varphi_{qg}, \gamma_{qg}) = x_i \cos \varphi_{qg} + y_i \sin \varphi_{qg} + \gamma_{qg} - r_i.$$

*Phi-функція для опуклого багатокутника і півплощини  $P_{qg}$ .* Phi-функцію для опуклого багатокутника  $K_i(u_i)$ , заданого його вершинами  $p_{ik} = (p_{ik}^x, p_{ik}^y)$ ,  $k = 1, \dots, l_i$ , і півплощини  $P_{qg}$  визначено в [87], вона має вигляд

$$\Phi^{K_i P_{qg}}(u_i, \varphi_{qg}, \gamma_{qg}) = \min_{k=1, \dots, l_i} (p_{ik}^x \cos \varphi_{qg} + p_{ik}^y \sin \varphi_{qg} + \gamma_{qg}).$$

*Включення кластерів об'єктів у контейнер.* Виведемо phi-функцію, яка аналітично подає обмеження включення (2.10),  $\widehat{\Lambda}_q(z_q) \subset \Omega$   
 $\Leftrightarrow \text{int } \widehat{\Lambda}_q(z_q) \cap \Omega^* = \emptyset$  для кожного  $q \in I_N$ , де  $\Omega^* = R^2 \setminus \text{int } \Omega$ .

Нехай  $\Phi^{T_i \Omega^*}(u_i)$  – phi-функція об'єктів  $T_i(u_i)$  та  $\Omega^*$ ,  $i \in \Xi_q$ .

*Твердження 2.2.* Неперервна і всюди визначена функція

$$\Phi^{\widehat{\Lambda}_q \Omega^*}(z_q) = \min \{ \Phi^{T_i \Omega^*}(u_i), i \in \Xi_q \} \quad (2.13)$$

є phi-функцією для опуклої оболонки  $\widehat{\Lambda}_q(z_q)$  та об'єкта  $\Omega^*$ .

*Доведення.* Покажемо, що якщо  $\widehat{\Lambda}_q(z_q) \subset \Omega$ , то  $T_i \subset \Omega$

$$\Leftrightarrow \text{int } T_i(u_i) \cap \Omega^* = \emptyset, i \in \Xi_q.$$

Припускаємо, що  $\Omega^* = \Omega_1^* \cup \Omega_2^* \cup \Omega_3^* \cup \Omega_4^*$ , де  $\Omega_s^* = \{(x, y) \mid g_s(x, y) \leq 0\}$  – півплощина,  $s = 1, 2, 3, 4$ ,  $g_1(x, y) = x_i$ ,  $g_2(x, y) = y$ ,  $g_3(x, y) = -x + l$ ,  $g_4(x, y) = -y + w$ ,

Для кожної півплощини  $\Omega_s^*$  можемо зробити висновок, що

1)  $\widehat{\Lambda}_q(z_q) \cap \Omega_s^* = \emptyset \Leftrightarrow \Lambda_q(z_q) \cap \Omega_s^* = \emptyset$  і тому  $T_i(u_i) \cap \Omega_s^* = \emptyset$ , для всіх  $i \in \Xi_q$ ;

2)  $\text{int } \widehat{\Lambda}_q \cap \text{int } \Omega_s^* \neq \emptyset \Leftrightarrow \text{int } \Lambda_q(z_q) \cap \text{int } \Omega_s^* \neq \emptyset$  і, отже, існує принаймні один об'єкт  $T_i(u_i)$  такий, що  $\text{int } T_i(u_i) \cap \text{int } \Omega_s^* \neq \emptyset$ ,  $i \in \Xi_q$ ;

3)  $(\text{int } \widehat{\Lambda}_q(z_q) \cap \text{int } \Omega_s^* = \emptyset) \wedge (\text{fr } \widehat{\Lambda}_q \cap \text{fr } \Omega_s^* \neq \emptyset) \Leftrightarrow$   
 $(\text{int } \Lambda_q(z_q) \cap \text{int } \Omega_s^* = \emptyset) \wedge (\text{fr } \Lambda_q(z_q) \cap \text{fr } \Omega_s^* \neq \emptyset)$  і тому  
 $\text{int } T_i(u_i) \cap \text{int } \Omega_s^* = \emptyset$  для всіх  $i \in \Xi_q$ , та існує принаймні один об'єкт  $T_i(u_i)$  такий, що  $\text{fr } T_i(u_i) \cap \text{fr } \Omega_s^* \neq \emptyset$ ,  $i \in \Xi_q$ .

Отже,  $\widehat{\Lambda}_q(z_q) \subset \Omega \Leftrightarrow \text{int } T_i(u_i) \cap \Omega^* = \emptyset$ ,  $i \in \Xi_q$ . Це означає, що  $\Phi^{\widehat{\Lambda}_q \Omega^*}(z_q) \geq 0$ , якщо і тільки якщо  $\Phi^{T_i \Omega^*}(u_i) \geq 0$  для всіх  $i \in \Xi_q$ .

Тепер сформулюємо за допомогою  $\phi$ -функцій умови включення в контейнер  $\Omega$  об'єктів  $T_i(u_i)$ ,  $i \in \Xi_q$ , таких форм: еліпса  $E_i(u_i)$ , кола  $C_i(v_i)$ , опуклого багатокутника  $K_i(u_i)$ .

*Phi-функція для включення еліпса.* Phi-функція для еліпса  $E(u_i)$ , заданого його півосями  $(a_i, b_i)$ , і об'єкта  $\Omega^*$  може бути визначена у вигляді

$$\Phi^{E_i \Omega^*}(u_i) = \min_{s=1, \dots, 4} g_{is}(u_i),$$

$$g_{i1}(u_i) = x_i - \sqrt{b_i^2 + (a_i^2 - b_i^2) \cos^2 \theta_i},$$

$$g_{i2}(u_i) = y_i - \sqrt{b_i^2 + (a_i^2 - b_i^2) \sin^2 \theta_i},$$

$$g_{i3}(u_i) = l - x_i - \sqrt{b_i^2 + (a_i^2 - b_i^2) \cos^2 \theta_i},$$

$$g_{i4}(u_i) = w - y_i - \sqrt{b_i^2 + (a_i^2 - b_i^2) \sin^2 \theta_i}.$$

Нерівність  $\Phi^{E_i\Omega^*}(u_i) \geq 0$  гарантує, що  $E_i(u_i) \subset \Omega$ . Визначимо  $\Phi$ -функцію на основі ідеї неперетину еліпса та півплощини, введеної в [137].

*Phi-функція для включення кола.*  $\Phi$ -функція для кола  $C(u_i)$ , заданого його радіусом  $r_i$ , і об'єкта  $\Omega^*$  має вигляд

$$\Phi^{C_i\Omega^*}(u_i) = \min_{s=1,\dots,4} f_{is}(u_i),$$

$$g_{i1}(u_i) = x_i - r_i, \quad g_{i2}(u_i) = y_i - r_i, \quad g_{i3}(u_i) = -x_i + l - r_i, \quad g_{i4}(u_i) = -y_i + w - r_i.$$

Нерівність  $\Phi^{C_i\Omega^*}(u_i) \geq 0$  гарантує, що  $C_i(v_i) \subset \Omega$ .

*Phi-функція для включення опуклого багатокутника.*  $\Phi$ -функція для опуклого багатокутника  $K(u_i)$  заданого його вершинами  $p_{ik} = (p_{ik}^x, p_{ik}^y)$ ,  $k = 1, \dots, l_i$ , і об'єкта  $\Omega^*$  має вигляд

$$\Phi^{K_i\Omega^*}(u_i) = \min_{\substack{k=1,\dots,l_i \\ s=1,\dots,4}} g_s(p_{ik}^x, p_{ik}^y).$$

Нерівність  $\Phi^{K_i\Omega^*}(u_i) \geq 0$  гарантує, що  $K_i(u_i) \subset \Omega$ .

#### 2.4. Математична модель задачі компонування кластерів об'єктів

Математична модель для розміщення кластерів об'єктів у прямокутній зоні  $\Omega$  можна сформулювати у формі

$$\max_{(u, \phi, \psi, \rho) \in W \subset R^\sigma} \rho, \quad (2.14)$$

$$W = \{(u, \phi, \psi, \rho) \in R^\sigma : \Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) - 0,5\rho \geq 0, q > g \in I_N, \rho \geq 0, \quad (2.15)$$

$$\Phi^{T_i T_j}(u_i, u_j, u_{ij}) \geq 0, i > j, (i, j) \in \Xi_q \times \Xi_q, q \in I_N, \Phi^{T_i \Omega^*}(u_i) \geq 0, i \in I_n\},$$

де  $\rho$  розглядають як відстань між опуклими оболонками  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$ ,  $u = (u_1, u_2, \dots, u_n)$  – вектор параметрів розміщення,  $u_i = (v_i, \theta_i)$ ,  $v_i = (x_i, y_i)$ ,  $\psi = (\varphi_{qg}, \gamma_{qg}, q > g \in J_N)$  є вектором допоміжних змінних у квазі-phi-функції опуклих оболонок  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$ ,  $\phi = (u_{ij}, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$  є вектором допоміжних змінних у квазі-phi-функції об'єктів  $T_i(u_i)$  і  $T_j(u_j)$ ,  $\Phi^{T_i T_j}(u_i, u_j, u_{ij})$  є квазі-phi-функцією об'єктів  $T_i(u_i)$  і  $T_j(u_j)$ ,  $\Phi^{T_i \Omega^*}(u_i)$  є phi-функцією об'єктів  $T_i(u_i)$  і  $\Omega^*$ , визначеної в (2.13),  $\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$  є квазі-phi-функцією опуклих оболонок  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$ , визначеної в (2.11),  $\sigma = 1 + 3n + \frac{\tau_q}{2} \sum_{q=1}^N m_q(m_q - 1) + N(N - 1)$  – кількість змінних задачі.  $\tau_q$  – кількість допоміжних змінних для функції, яка описує обмеження неперетину для пари об'єктів  $T_i(u_i)$  і  $T_j(u_j)$ ,  $(i, j) \in \Xi_q \times \Xi_q, q \in I_N$ . Зокрема,  $\tau_q = 0$  для двох кіл,  $\tau_q = 1$  для двох еліпсів,  $\tau_q = 2$  для двох опуклих багатокутників (див. п. 2.3).

У випадку двох кіл  $C_i(u_i)$  і  $C_j(u_j)$  використовується phi-функція  $\Phi^{C_i C_j}(u_i, u_j)$  замість квазі-phi-функції  $\Phi^{T_i T_j}(u_i, u_j, u_{ij})$ .

Множина  $W$ , задана (2.15), визначається системою нерівностей з диференційованими функціями. Модель (2.14), (2.15) є неопуклою і неперервною задачею нелінійного програмування. Це точне формулювання в

тому сенсі, що вона містить всі оптимальні розв'язки.

Модель (2.14), (2.15) містить  $O(\sum_{q=1}^N m_q^2) + O(N^2)$  нелінійних нерівностей

та  $O(\sum_{q=1}^N m_q^2) + O(N^2)$  змінних завдяки допоміжним змінним у квазі-phi-функціях.

Розглянемо ефективний підхід, який використовує алгоритм побудови допустимих стартових розв'язків для знаходження локальних екстремумів задачі (2.14), (2.15).

## 2.5. Алгоритм розв'язання задачі компонування кластерів об'єктів

Стратегія розв'язання заснована на багатостартовому (multistart) алгоритмі і містить три основні етапи: 1. Формування набору можливих вихідних точок для задачі (2.14), (2.15). 2. Пошук набору локальних максимумів для задачі (2.14), (2.15), починаючи з кожної допустимої точки, отриманої на етапі 1. 3. Вибір найбільшого локального максимуму зі знайдених на етапі 2.

Для формування допустимих стартових точок задачі (2.14), (2.15) розроблено спеціальний алгоритм (алгоритм допустимих стартових точок), який містить такі кроки.

Крок 1. З кожним кластером  $\Lambda_q(z_q)$  асоціюється окіл  $\lambda C_q$  зі змінною центральною точкою  $v_q = (x_q, y_q)$  і змінним радіусом  $\lambda R_q$ , де  $\lambda$  є параметром

масштабування (scaling parameter),  $R_q = \sqrt{\sum_{i \in \Xi_q} S_i}$ ,  $S_i$  – площа об'єкта  $T_i$ ,

$i \in \Xi_q, q \in I_N$ .

Крок 2. Окіл  $\lambda C_q$ ,  $q \in I_N$ , збільшується в межах прямокутного контейнера  $\Omega$  якомога більше за допомогою розв'язання такої підзадачі нелінійного програмування:

$$\max_{(v, \lambda) \in V} \lambda, \quad (2.16)$$

$$V = \{(v, \lambda) \in R^{2N+1} : \Phi^{C_q C_g}(v_q, v_g, \lambda) \geq 0, q > g \in I_N, \Phi^{C_q \Omega^*}(v_q, \lambda) \geq 0, q \in I_N, \lambda \geq 0\}, \quad (2.17)$$

де  $v = (v_1, v_2, \dots, v_N)$  – вектор змінних параметрів розміщення колових зон  $\lambda C_q$ ,  $q \in I_N$ ;  $\Phi^{C_q C_g}(v_q, v_g, \lambda)$  є phi-функцією зон  $\lambda C_q$  і  $\lambda C_g$  із центрами у точках  $v_q = (x_q, y_q)$  і  $v_g = (x_g, y_g)$  та відповідними радіусами  $\lambda R_q$  і  $\lambda R_g$ ,  $q > g \in I_N$ ,

$$\Phi^{C_q C_g}(v_q, v_g, \lambda) = (x_q - x_g)^2 + (y_q - y_g)^2 - (\lambda R_q + \lambda R_g)^2;$$

$\Phi^{C_q \Omega^*}(v_q, \lambda)$  – phi-функція колової зони  $\lambda C_q$  та об'єкта  $\Omega^* = R^2 \setminus \text{int } \Omega$ ,  $q \in I_N$ ,

$$\Phi^{C_q \Omega^*}(v_q, \lambda) = \min_{s=1, \dots, 4} f_{qs}(v_q, \lambda),$$

$$f_{q1}(v_q, \lambda) = x_q - \lambda R_q, f_{q2}(v_q, \lambda) = y_q - \lambda R_q, f_{q3}(v_q, \lambda) = l - x_q - \lambda R_q,$$

$$f_{q4}(v_q, \lambda) = w - y_q - \lambda R_q.$$

Шукаємо локальний максимум задачі (2.16), (2.17), починаючи з допустимої точки  $(v^0 = (v_1^0, v_2^0, \dots, v_N^0), \lambda^0 = 0)$ , де  $v_q^0 \in \Omega$ ,  $q \in I_N$  – випадково сформовані точки. Позначимо локальну точку максимуму задачі (2.16), (2.17) як  $(v^*, \lambda^*)$ .

Крок 3. Визначаються початкові значення змінних  $\varphi_{qg}^0, \gamma_{qg}^0$  (отримані за допомогою тривіальних геометричних побудов) для кожної квазі-phi-функції  $\Phi^{\hat{\Lambda}_q \hat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$ ,  $q > g \in I_N$  у задачі (2.14), (2.15). Ці змінні

розглядають як параметри відокремлювальних прямих між кожною парою колових зон  $\lambda^* C_q$  і  $\lambda^* C_g$  радіусами  $\lambda^* R_q$  і  $\lambda^* R_g$  з центрами у точках  $v_q^*$  та  $v_g^*$ ,  $q > g \in I_N$ .

Крок 4. Знаходять допустимі точки задачі (2.14), (2.15) за допомогою розв'язання такої підзадачі нелінійного програмування:

$$\max_{(u, \phi, \psi, \beta) \in G} \beta, \quad (2.18)$$

$$G = \{(u, \phi, \psi, \beta) \in R^\sigma : \Phi^{\hat{\Lambda}_q \hat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}, \beta) \geq 0, q > g \in I_N, 0 \leq \beta \leq 1, (2.19)$$

$$\Phi^{T_i T_j}(u_i, u_j, u_{ij}, \beta) \geq 0, i > j, (i, j) \in \Xi_q \times \Xi_q, q \in I_N, \Phi^{T_i \Omega^*}(u_i, \mu) \geq 0, i \in I_n\},$$

де  $u = (u_1, u_2, \dots, u_n)$ ,  $u_i = (v_i, \theta_i)$ ,  $v_i = (x_i, y_i)$ ,  $\psi = (\varphi_{qg}, \gamma_{qg}, q > g \in J_N)$ ;  $\beta$  є

параметром масштабування об'єктів;  $\phi = (u_{ij}, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$  є

вектором допоміжних змінних у квазі-phi-функції для об'єктів  $\beta T_i(u_i)$  і

$\beta T_j(u_j)$ ;  $\Phi^{T_i T_j}(u_i, u_j, u_{ij}, \beta)$  – квазі-phi-функція об'єктів  $\beta T_i(u_i)$  і  $\beta T_j(u_j)$ ;

$\Phi^{T_i \Omega^*}(u_i)$  – phi-функція об'єктів  $\beta T_i(u_i)$  та  $\Omega^*$ ;  $\Phi^{\hat{\Lambda}_q \hat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}, \beta)$

– квазі-phi-функція для двох кластерів, визначених у (2.11);

$\sigma = 1 + 3n + \frac{\tau_q}{2} \sum_{q=1}^N m_q(m_q - 1) + N(N - 1)$  – кількість змінних задачі,  $\tau_q$  –

кількість допоміжних змінних.

Використовуємо допустиму стартову точку  $(u^0, \phi^0, \psi^0, \beta^0 = 0)$  для розв'язання задачі (2.18), (2.19), де  $\psi^0 = (\varphi_{qg}^0, \gamma_{qg}^0, q > g \in J_N)$  отримано на

кроці 3;  $v_i^0 = (x_i^0, y_i^0) \in C_q$  – вектор випадково сформованих параметрів

трансляції і  $\theta_i^0 \in [0, 2\pi]$  є випадково згенерованим параметром обертання

об'єкта  $\beta T_i(u_i)$  для  $i \in \Xi_q, q \in I_N$ ;  $\phi^0 = (u_{ij}^0, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$ , вектор  $u_{ij}^0$

допоміжних змінних знайдено з тривіальних геометричних обчислень залежно

від форми об'єкта (крім кіл), так що  $\Phi^{T_i T_j}(u_i^0, u_j^0, u_{ij}^0, \beta^0 = 0) \geq 0$ . Зокрема, для двох еліпсів  $u_{ij}^0 = \phi_{ij}^0$ , для двох опуклих багатокутників  $u_{ij}^0 = (\phi_{ij}^0, \gamma_{ij}^0)$  (див. п. 2.3).

Глобальний максимум задачі (2.18), (2.19), (і.е.  $\beta^* = 1$ ), забезпечує допустимий розв'язок задачі (2.14), (2.15). Точку глобального максимуму можна використовувати як стартову точку для пошуку локального максимуму задачі (2.14), (2.15).

Застосування алгоритму декомпозиції, описаного в [25], рекомендовано для розв'язання задачі (2.14), (2.15), коли  $m_q \geq 11$  для кластерів  $\Lambda_q(z_q)$ ,  $q \in J_N$ , з метою зменшення обчислювальних витрат (часу та пам'яті).

## 2.6. Приклади компонування кластерів у прямокутному контейнері

Наведемо деякі приклади, щоб показати ефективність запропонованого підходу. Експерименти виконано на комп'ютері AMD FX (tm)-6100, 3,30 ГГц, мовою програмування C ++, Windows 7. Для локальної оптимізації використано код IPOPT (<https://projects.coin-or.org/Ipoprt>), розроблений у [127]. Для запуску цього програмного забезпечення використовувались параметри за замовчуванням.

Нижче наведено приклади для задачі компонування в прямокутному контейнері  $\Omega = \{(x, y) \in R^2 : 0 \leq x \leq l = 15, 0 \leq y \leq w = 15\}$ . Для кожного прикладу програму запускали 100 разів.

Приклад 2.3. Задано впорядкований набір  $n = 30$  еліпсів. Набір еліпсів розділений на  $N=3$  кластери  $\Lambda_q(z_q)$ ,  $q = 1, 2, 3$ :  $m_1 = 10, m_2 = 10, m_3 = 10$ .

Усі еліпси визначаються їх розмірами  $\{(a_i, b_i), i=1, \dots, 30\} = \{(2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5), (1.5 \ 1.2), (1.3 \ 1.0), (1.2 \ 0.9), (1.1 \ 0.8), (1.0 \ 0.75), (0.9 \ 0.6), (0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5), (1.5 \ 1.2), (1.3 \ 1.0), (1.2 \ 0.9), (1.1 \ 0.8), (1.0$

0.75), (0.9 0.6), (0.8 0.5), (0.7 0.3), (2.0 1.5), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8), (1.0 0.75)}.

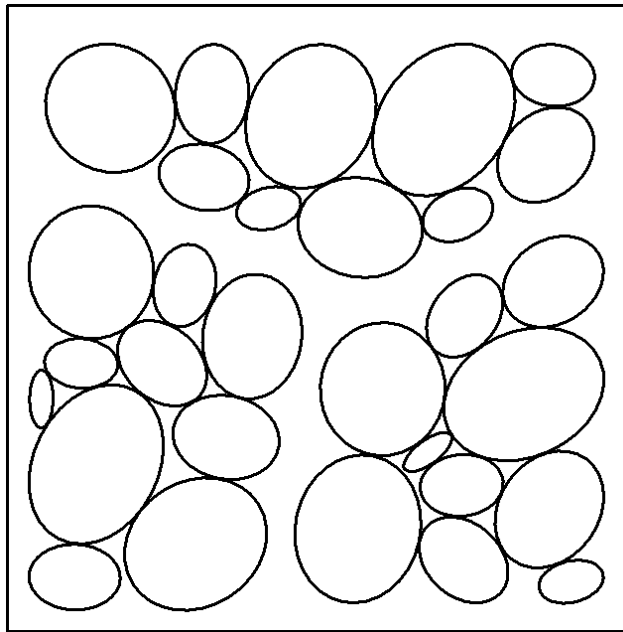
Результат розв'язання:

Параметри розміщення еліпсів

$\{(x_i, y_i, \theta_i), i=1, \dots, 30\} = \{(10.469949 \ 12.267537 \ 2.244218), (7.279733 \ 12.352075 \ 2.063582), (2.471444 \ 12.548774 \ 13.249926), (8.464111 \ 9.691723 \ 0.145847), (12.920038 \ 11.433172 \ 5.530417), (4.909339 \ 12.890715 \ 1.640578), (4.716335 \ 10.894489 \ 0.195532), (13.092110 \ 13.335890 \ 6.393843), (10.806375 \ 9.993924 \ -0.401328), (6.267221 \ 10.149879 \ 6.032813), (0.813462 \ 5.582920 \ 1.582319), (2.129866 \ 4.016161 \ 2.043699), (4.516627 \ 2.106510 \ 5.713416), (2.014535 \ 8.617626 \ 4.606157), (5.875381 \ 7.080877 \ 1.710554), (5.246435 \ 4.663067 \ 0.181324), (3.712908 \ 6.439680 \ 3.845887), (1.613133 \ 1.313706 \ 6.313737), (4.250763 \ 8.301450 \ 4.942235), (1.768064 \ 6.435263 \ 0.065954), (13.527926 \ 1.206001 \ 2.906999), (10.087641 \ 4.308896 \ 2.501728), (12.390656 \ 5.699737 \ 2.716022), (8.409176 \ 2.472279 \ 4.931800), (9.003824 \ 5.818192 \ 1.638294), (13.004545 \ 2.937130 \ 5.336177), (13.095024 \ 8.392657 \ -0.572537), (10.944860 \ 1.726727 \ 0.720864), (10.969680 \ 7.572605 \ 5.373848), (10.898378 \ 3.513018 \ -0.101262)}.$

Значення цільової функції становить  $\rho^* = 2,109291$ , час обчислення доівнює 7131,446 секунд.

Відповідне оптимізоване компонування наведено на рис. 2.3.



**Рис. 2.3.** Локально-оптимальне розміщення кластерів у прикладі 2.3

Приклад 2.4. Задано впорядкований набір  $n=36$  кіл. Набір кіл розділений на  $N=4$  кластери  $\Lambda_q(z_q)$ ,  $q=1, 2, 3, 4$ :  $m_1=10$ ,  $m_2=10$ ,  $m_3=8$ ,  $m_4=8$ .

Усі кола визначаються радіусами:

$\{(r_i), i=1, \dots, 36\} = \{(1.495), (1.500), (1.501), (1.206), (1.009), (0.912), (0.815), (0.765), (0.621), (0.524), (0.336), (1.550), (1.533), (1.512), (1.239), (1.042), (0.945), (0.848), (0.7925), (0.654), (0.557), (0.380), (1.605), (1.566), (1.523), (1.272), (1.075), (0.978), (0.881), (0.820), (0.687), (0.590), (0.424), (1.660), (1.599), (1.534)\}$ .

Результат розв'язання:

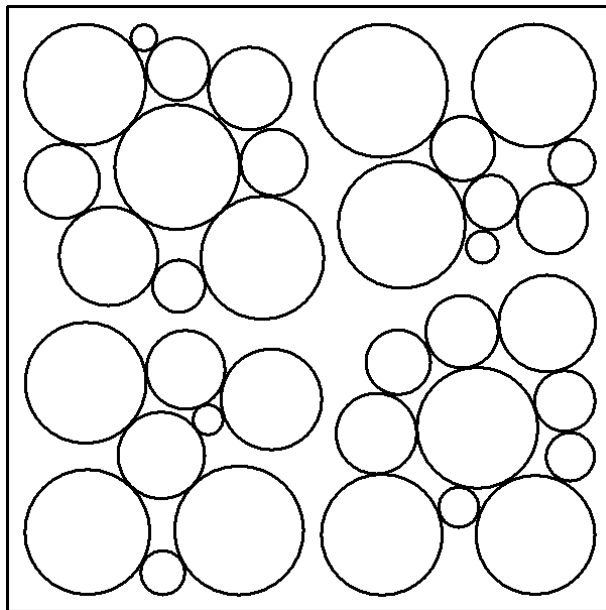
Параметри розміщення кіл

$\{(x_i, y_i), i=1, \dots, 36\} = \{(13.089089 \quad 1.910911), (9.282866 \quad 1.915911), (11.652953 \quad 4.566297), (13.378089 \quad 7.152616), (9.147148 \quad 4.421237), (11.269926 \quad 6.948703), (9.688888 \quad 6.189878), (13.819089 \quad 5.231585), (13.954597 \quad 3.846912), (11.189529 \quad 2.595038), (3.381819 \quad 14.237068), (4.209528 \quad 11.030014), (6.318116 \quad 8.780847), (1.927911 \quad 13.072089), (2.506044 \quad 8.821697), (6.003824 \quad 12.979437), (1.360911 \quad 10.681408), (6.604792 \quad 11.144527), (4.222026 \quad 13.460604), (4.247002$

8.078382), (3.849810 0.972911), (4.971098 4.766312), (5.740826 2.020911), (1.981911 1.981911), (1.938911 5.681867), (6.544523 5.269738), (3.814234 3.883877), (4.418702 6.006885), (13.509939 9.758354), (11.287589 11.489670), (11.993830 10.158402), (13.994089 11.147396), (11.771968 9.049173), (9.264512 12.924089), (9.777002 9.600308), (13.050089 13.050089)}.

Значення цільової функції становить  $\rho^* = 0,415912$ , час обчислення дорівнює 484,461 секунд.

Відповідне оптимізоване компонування наведено на рис. 2.4.



**Рис. 2.4.** Локально-оптимальне розміщення кластерів у прикладі 2.4

Приклад 2.5. Задано впорядкований набір  $n = 36$  об'єктів – еліпсів та кіл. Набір об'єктів поділяється на  $N=4$  кластерів  $\Lambda_q(z_q)$ ,  $q = 1, 2, 3, 4$ :  $m_1 = 10$  кіл,  $m_2 = 10$  кіл,  $m_3 = 8$  еліпсів,  $m_4 = 8$  еліпсів.

Усі кола визначаються радіусами:

$\{(r_i), i=1, \dots, 20\} = \{(1.5), (1.5), (1.5), (1.2), (1.0), (0.9), (0.8), (0.75), (0.6), (0.5), (0.3), (1.5), (1.5), (1.5), (1.2), (1.0), (0.9), (0.8), (0.75), (0.6)\}$ .

Усі еліпси визначаються розмірами:

$\{(a_i, b_i), i=21, \dots, 36\} = \{(0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5), (1.5 \ 1.2), (1.3 \ 1.0), (1.2 \ 0.9), (1.1 \ 0.8), (1.0 \ 0.75), (0.9 \ 0.6), (0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5)\}$ .

Результат розв'язання:

Параметри розміщення кіл:

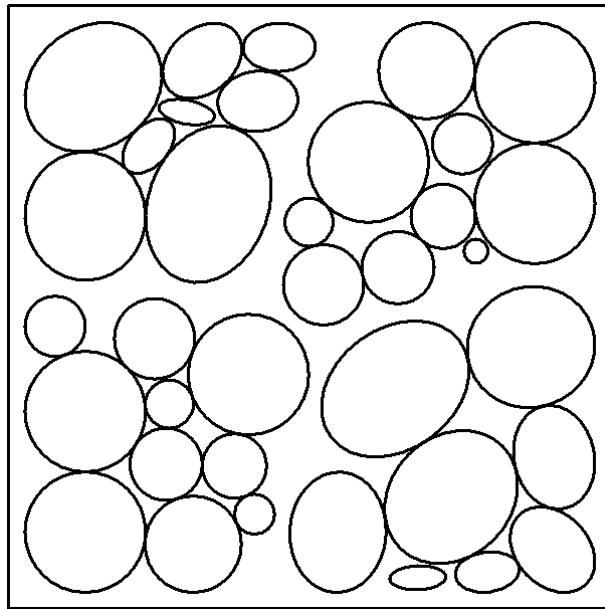
$\{(x_i, y_i), i=1, \dots, 20\} = \{(5.969333, 5.831559), (1.910221, 4.910224), (1.910221, 1.910223), (4.593503, 1.610221), (3.633171, 6.721701), (3.920715, 3.599532), (5.620213, 3.558210), (1.160222, 7.031545), (4.001869, 5.097335), (6.120123, 2.358173), (11.629726, 8.900163), (13.089779, 13.089779), (13.089779 \ 10.089779), (8.950916, 11.115733), (10.406497, 13.389779), (7.836995, 8.063426), (9.688425, 8.490281), (10.812773, 9.765367), (11.294872, 11.568861), (7.474932, 9.621922)\}$ .

Параметри розміщення еліпсів:

$\{(x_i, y_i, \theta_i), i=21, \dots, 36\} = \{(11.912729 \ 0.916670 \ 9.295412), (10.183793 \ 0.775760 \ -0.039353), (9.620035 \ 5.473407 \ -0.639827), (11.012582 \ 2.782653 \ -3.922661), (12.998276 \ 6.515842 \ -3.441852), (8.195878 \ 1.909350 \ 1.627631), (13.574899 \ 3.775017 \ 13.927174), (13.533974 \ 1.475715 \ 3.943306), (4.818922 \ 13.635480 \ -0.705039), (6.200920 \ 12.630733 \ 3.035222), (6.736535 \ 13.971098 \ -0.026565), (3.486483 \ 11.514966 \ 5.442422), (4.430982 \ 12.359577 \ 0.186450), (4.973013 \ 10.072013 \ -1.233481), (2.113962 \ 12.981281 \ 2.518350), (1.910433 \ 9.773977 \ 17.233423)\}$ .

Значення цільової функції становить  $\rho^* = 0,410222$ , час обчислення дорівнює 1344,9 секунд.

Відповідне оптимізоване компонування наведено на рис. 2.5.



**Рис. 2.5.** Локально-оптимальне розміщення кластерів у прикладі 2.5

Приклад 2.6. Задано впорядкований набір  $n = 36$  об'єктів. Набір об'єктів поділяється на  $N=4$  кластерів  $\Lambda_q(z_q)$ ,  $q = 1, 2, 3, 4$ ;  $m_1 = 10$  кіл,  $m_2 = 10$  кіл,  $m_3 = 8$  еліпсів,  $m_4 = 8$  еліпсів.

Усі кола визначаються радіусами:

$$\{(r_i), i=1, \dots, 20\} = \{(1.625), (1.575), (1.525), (1.275), (1.075), (0.975), (0.875), (0.8125), (0.675), (0.575), (0.400), (1.625), (1.575), (1.525), (1.275), (1.075), (0.975), (0.875), (0.8125), (0.675)\}.$$

Усі еліпси визначаються розмірами:

$$\{(a_i, b_i), i=21, \dots, 36\} = \{(0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5), (1.5 \ 1.2), (1.3 \ 1.0), (1.2 \ 0.9), (1.1 \ 0.8), (1.0 \ 0.75), (0.9 \ 0.6), (0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5)\}.$$

Результат розв'язання:

Параметри розміщення кіл:

$$\{(x_i, y_i), i=1, \dots, 20\} = \{(5.025933 \ 1.876324), (1.826324 \ 1.826324), (3.974219 \ 6.518053), (6.563760 \ 7.583078), (3.350845 \ 3.993889), (1.451364$$

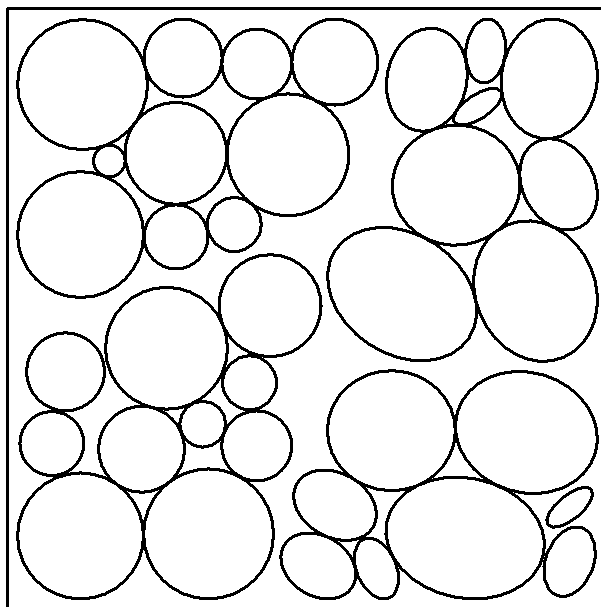
5.929808), (6.228908 4.078766), (1.109501 4.136463), (6.051897 5.654399), (4.881146 4.620133), (2.547234 11.199363), (1.876324 13.109992), (1.826324 9.360637), (7.011083 11.353004), (4.211329 11.390181), (8.183466 13.673676), (4.390190 13.773676), (6.234106 13.623755), (4.213121 9.302682), (5.668491 9.610177)}.

Параметри розміщення еліпсів:

$\{(x_i, y_i, \theta_i), i=21, \dots, 36\} = \{(11.954503 \ 13.951223 \ 14.239492), (11.755667 \ 12.554007 \ 2.556380), (9.866202 \ 7.871985 \ -11.985510), (13.200316 \ 7.946103 \ 4.316218), (11.218788 \ 10.594559 \ -25.265057), (13.543329 \ 13.252969 \ -23.435600), (10.483036 \ 13.231361 \ 1.774510), (13.776518 \ 10.616527 \ 16.797305), (8.189442 \ 2.594354 \ -21.480649), (7.771967 \ 1.076809 \ 22.539607), (14.052522 \ 1.182705 \ 20.790155), (9.223332 \ 1.011216 \ -8.266075), (14.056688 \ 2.541830 \ -19.537970), (11.441889 \ 1.774003 \ -12.367135), (12.971454 \ 4.412810 \ -15.416986), (9.596560 \ 4.458588 \ 44.002983)\}$ .

Значення цільової функції становить  $\rho^* = 0,251325$ , час обчислення дорівнює 1285,73 секунд.

Відповідне оптимізоване компонування наведено на рис. 2.6.



**Рис. 2.6.** Локально-оптимальне розміщення кластерів у прикладі 2.6

Приклад 2.7. Задано впорядкований набір  $n = 36$  об'єктів. Набір об'єктів поділяється на  $N=4$  кластерів  $\Lambda_q(z_q)$ ,  $q = 1, 2, 3, 4$ :  $m_1 = 10$  опуклих багатокутників,  $m_2 = 10$  кіл,  $m_3 = 8$  еліпсів,  $m_4 = 8$  еліпсів.

Усі багатокутники визначаються вершинами:

$$\{(p_{ik}^x, p_{ik}^y), k = 1, \dots, 6, i = 1, \dots, 10\} = \{(1.975377 \ 0.234652), (0.716736 \ 1.400371), (-1.258641 \ 1.165719), (-1.975377 \ -0.234652), (-0.716736 \ -1.400371), (1.258641 \ -1.165719)\}, \{(1.711902 \ 0.463525), (0.374241 \ 1.467221), (-1.337661 \ 1.003696), (-1.711902 \ -0.463525), (-0.374241 \ -1.467221), (1.337661 \ -1.003696)\}, \{(1.425610 \ 0.680986), (0.083738 \ 1.497944), (-1.341873 \ 0.816959), (-1.425610 \ -0.680986), (-0.083738 \ -1.497944), (1.341873 \ -0.816959)\}, \{(1.213525 \ 0.705342), (-0.156793 \ 1.193426), (-1.370318 \ 0.488084), (-1.213525 \ -0.705342), (0.156793 \ -1.193426), (1.370318 \ -0.488084)\}, \{(0.919239 \ 0.707107), (-0.336465 \ 0.965926), (-1.255704 \ 0.258819), (-0.919239 \ -0.707107), (0.336465 \ -0.965926), (1.255704 \ -0.258819)\}, \{(0.705342 \ 0.728115), (-0.488084 \ 0.822191), (-1.193426 \ 0.094076), (-0.705342 \ -0.728115), (0.488084 \ -0.822191), (1.193426 \ -0.094076)\}, \{(0.499390 \ 0.712805), (-0.599103 \ 0.670936), (-1.098492 \ -0.041869), (-0.499390 \ -0.712805), (0.599103 \ -0.670936), (1.098492 \ 0.041869)\}, \{(0.309017 \ 0.713292), (-0.669131 \ 0.557359), (-0.978148 \ -0.155934), (-0.309017 \ -0.713292), (0.669131 \ -0.557359), (0.978148 \ 0.155934)\}, \{(0.140791 \ 0.592613), (-0.699431 \ 0.377592), (-0.840222 \ -0.215021), (-0.140791 \ -0.592613), (0.699431 \ -0.377592), (0.840222 \ 0.215021)\}, \{(0.062767 \ 0.498459), (-0.659301 \ 0.283203), (-0.722068 \ -0.215256), (-0.062767 \ -0.498459), (0.659301 \ -0.283203), (0.722068 \ 0.215256)\}.$$

Усі кола визначаються радіусами:

$$\{r_i, i=11, \dots, 20\} = \{0.336, 1.55, 1.533, 1.512, 1.239, 1.042, 0.945, 0.848, 0.7925, 0.654\}$$

Усі еліпси визначаються розмірами:

$\{(a_i, b_i), i=21, \dots, 36\} = \{(0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5), (1.5 \ 1.2), (1.3 \ 1.0), (1.2 \ 0.9), (1.1 \ 0.8), (1.0 \ 0.75), (0.9 \ 0.6), (0.8 \ 0.5), (0.7 \ 0.3), (2.0 \ 1.5), (1.8 \ 1.5), (1.6 \ 1.5)\}$ .

Результат розв'язання:

Параметри розміщення багатокутників:

$\{(x_i, y_i, \theta_i), i=1, \dots, 10\} = \{(3.447806 \ 9.597127 \ 2.606907), (4.632826 \ 12.154203 \ 6.031453), (1.935707 \ 13.328641 \ 2.695960), (7.137488 \ 11.281372 \ 4.553086), (6.594090 \ 13.766083 \ 3.344861), (6.187148 \ 9.257646 \ -0.386618), (1.270813 \ 11.108185 \ 0.670004), (5.355596 \ 7.966335 \ 0.229233), (4.062287 \ 14.055842 \ 3.636614), (7.862803 \ 9.559287 \ 6.671758)\}$ .

Параметри розміщення кіл:

$\{(x_i, y_i), i=11, \dots, 20\} = \{(14.242145 \ 4.607830), (11.278663 \ 1.905800), (8.391616 \ 6.969211), (9.039026 \ 3.993831), (11.103865 \ 5.862139), (13.546072 \ 3.321421), (13.699200 \ 1.300800), (8.985716 \ 1.203801), (13.126975 \ 5.194957), (11.917846 \ 4.076973)\}$ .

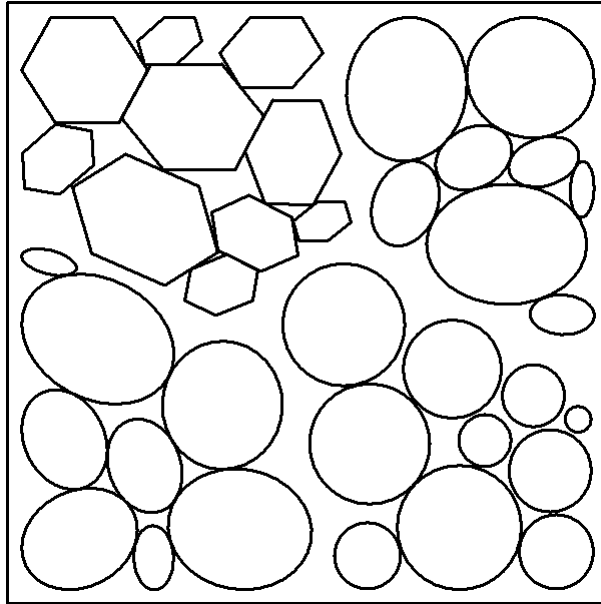
Параметри розміщення еліпсів:

$\{(x_i, y_i, \theta_i), i=21, \dots, 36\} = \{(3.646227 \ 1.155469 \ 1.533947), (1.038618 \ 8.531725 \ 0.246199), (2.256484 \ 6.610724 \ 0.489860), (5.799650 \ 1.865204 \ 6.453077), (5.363840 \ 4.958834 \ 1.597596), (1.798943 \ 1.623608 \ 5.811350), (1.423676 \ 4.102206 \ 7.386076), (3.423022 \ 3.445736 \ 4.374109), (9.931251 \ 9.996600 \ 5.101874), (11.636070 \ 11.136792 \ 2.665959), (13.383614 \ 11.035564 \ 2.838929), (13.844824 \ 7.219741 \ 6.333814), (14.343409 \ 10.348285 \ 1.605276), (12.459639 \ 8.974386 \ 0.003111), (9.963431 \ 12.848701 \ 1.699000), (13.051879 \ 13.13605 \ 10.285045)\}$ .

Значення цільової функції становить  $\rho^* = 0,355801$ , час обчислення дорівнює 4625,15 секунд.

Відповідне оптимізоване компонування наведено на рис. 2.7.

Приклад 2.8. Задано впорядкований набір  $n = 30$  об'єктів. Набір об'єктів поділяється на  $N=3$  кластерів  $\Lambda_q(z_q)$ ,  $q=1, 2, 3$ :  $m_1=10$  опуклих багатокутників,  $m_2=10$  кіл,  $m_3=10$  еліпсів.



**Рис. 2.7.** Локально-оптимальне розміщення кластерів у прикладі 2.7

Усі багатокутники визначаються вершинами:

$$\{(p_{ik}^x, p_{ik}^y), k=1, \dots, 6, i=1, \dots, 10\} = \{(2.000000 \ 0.000000), (1.000000 \ 1.732051), (-1.000000 \ 1.732051), (-2.000000 \ 0.000000), (-1.000000 \ -1.732051), (1.000000 \ -1.732051)\}, \{(1.800000 \ 0.000000), (0.900000 \ 1.558846), (-0.900000 \ 1.558846), (-1.800000 \ 0.000000), (-0.900000 \ -1.558846), (0.900000 \ -1.558846)\}, \{(1.700000 \ 0.000000), (0.850000 \ 1.472243), (-0.850000 \ 1.472243), (-1.700000 \ 0.000000), (-0.850000 \ -1.472243), (0.850000 \ -1.472243)\}, \{(1.600000 \ 0.000000), (0.800000 \ 1.385641), (-0.800000 \ 1.385641), (-1.600000 \ 0.000000), (-0.800000 \ -1.385641), (0.800000 \ -1.385641)\}, \{(1.400000 \ 0.000000), (0.700000 \ 1.212436), (-0.700000 \ 1.212436), (-1.400000 \ 0.000000), (-0.700000 \ -1.212436), (0.700000 \ -1.212436)\}, \{(1.300000 \ 0.000000), (0.650000 \ 1.125833), (-0.650000$$

1.125833), (-1.300000 0.000000), (-0.650000 -1.125833), (0.650000 -1.125833)}, {1.200000 0.000000), (0.600000 1.039230), (-0.600000 1.039230), (-1.200000 0.000000), (-0.600000 -1.039230), (0.600000 -1.039230)}, {(1.100000 0.000000), (0.550000 0.952628), (-0.550000 0.952628), (-1.100000 0.000000), (-0.550000 -0.952628), (0.550000 -0.952628)}, {(1.000000 0.000000), (0.500000 0.866025), (-0.500000 0.866025), (-1.000000 0.000000), (-0.500000 -0.866025), (0.500000 -0.866025)}, {(0.900000 0.000000), (0.450000 0.779423), (-0.450000 0.779423), (-0.900000 0.000000), (-0.450000 -0.779423), (0.450000 -0.779423)}.

Усі кола визначаються радіусами:

$\{r_i, i=11,\dots,20\} = \{0.336, 1.55, 1.533, 1.524, 1.513, 1.242, 1.23, 1.048, 0.951, 0.854\}$ .

Усі еліпси визначаються розмірами:

$\{(a_i, b_i), i=21,\dots,30\} = \{(0.8 0.5), (0.7 0.3), (2.0 1.5), (1.9 1.6), (1.8 1.5), (1.6 1.5), (1.5 1.2), (1.3 1.0), (1.2 0.9), (1.1 0.8)\}$ .

Результат розв'язання:

Параметри розміщення багатокутників:

$\{(x_i, y_i, \theta_i), i=1,\dots,10\} = \{(13.006728 7.204303 2.567940), (7.917914 11.613054 4.423250), (13.314400 11.008183 4.712389) (11.061231 13.271814 2.094380), (10.629721 10.486177 3.665191), (10.185459 8.006747 3.615137), (13.586642 13.747413 3.141592), (13.807690 4.598105 2.567940), (8.341396 9.221119 3.376054) (8.985800 14.007219 6.283185)\}$ .

Параметри розміщення кіл:

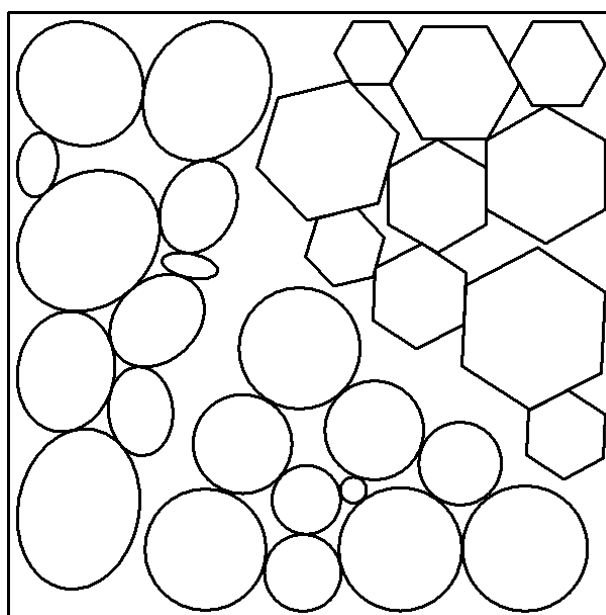
$\{(x_i, y_i), i=11,\dots,20\} = \{(8.537177 3.208783), (12.787093 1.765862), (9.700995 1.746357), (4.878381 1.737357), (7.206433 6.712881), (5.794319 4.347302), (9.055276 4.686597), (11.178919 3.862321), (7.286139 1.164357), (7.371932 2.967318)\}$ .

Параметри розміщення еліпсів:

$\{(x_i, y_i, \theta_i), i=21, \dots, 30\} = \{(0.720674 \ 11.248417 \ 4.850299), (4.490215 \ 8.752444 \ 3.338436), (1.734800 \ 2.740444 \ 1.764414), (1.981734 \ 9.367337 \ 5.537979), (4.912270 \ 13.064601 \ 2.125433), (1.778636 \ 13.250446 \ 3.779380), (1.425379 \ 6.137135 \ 4.902743), (3.677896 \ 7.396278 \ 2.407474), (4.702453 \ 10.211496 \ 2.031648), (3.268506 \ 5.159948 \ 1.494571)\}.$

Значення цільової функції становить  $\rho^* = 0,213358$ , час обчислення дорівнює 7986,627 секунд.

Відповідне оптимізоване компонування наведено на рис. 2.8.



**Рис. 2.8.** Локально-оптимальне розміщення кластерів у прикладі 2.8

## 2.7. Оптимізоване пакування кластерів за умови балансу

Часто об'єкти мають утворювати низку кластерів. Наприклад, у проблемі завантаження контейнерів [227] об'єкти кластеризовані відповідно до їх форм, типів, марок або кольорів. Об'єкти, що відповідають тому самому постачальнику або кінцевому пункту призначення (клієнту), також можуть бути згруповані в кластери, щоб допомогти особі, що приймає рішення, під час завантаження/розвантаження контейнера.

Кількість об'єктів, які потрібно розмістити у кластері, часто фіксується разом з їх конфігураціями. Однак загальну форму кластера у багатьох випадках не можна визначити, вона залежить від розміщення об'єктів всередині кластера. Форма кластера визначена у п. 2.2 як опукла оболонка об'єктів, що утворюють кластер. Оскільки загальноприйнятим є припущення, що об'єкти не перетинаються, форма кластера змінюється залежно від розміщення рухомих об'єктів у кластері.

У цьому підрозділі передбачено, що відомо кількість кластерів, кількість об'єктів у кластері, а також їх форми, розміри та маси. Компонування кластерів вважають допустимим, якщо: а) об'єкти у кластері не перетинаються; б) кластери не перетинаються і повністю належать прямокутному контейнеру; в) враховуються умови балансу. Щоб отримати дисперсну систему кластерів, максимізується відстань між кластерами. Цей підхід отримано на підставі вимоги мати більше простору між кластерами для спрощення їх завантаження/розвантаження.

Умови балансу системи важливі для задач завантаження контейнерів. Їх накладають для запобігання обертання контейнера під час переміщення транспортного засобу.

**Постановка задачі.** Нехай  $\Omega$  – прямокутний контейнер заданої довжини  $l$ , ширини  $w$  і маси  $w_0$ . І нехай  $T_i$ ,  $i \in \{1, 2, \dots, n\} = I_n$ , – впорядкований набір опуклих об'єктів. Кожен об'єкт  $T_i$  визначається його метричними характеристиками (розмірами) та масою  $w_i$ . Для того, щоб охарактеризувати положення об'єкта  $T_i$ , використовується вектор змінних параметрів розміщення  $u_i = (v_i, \theta_i)$ . Тут  $v_i = (x_i, y_i)$  – вектор трансляції, а  $\theta_i$  – кут обертання. Передбачено, що об'єкт зосереджений у початку власної системи координат. Переміщення об'єкта  $T_i$ , тобто його обертання на кут  $\theta_i$  і трансляція вектором  $v_i$  можуть бути подані як  $T_i(u_i) = \{p \in R^2 : p = v_i + M(\theta_i) \cdot p^0, \forall p^0 \in T_i^0\}$ , де  $T_i^0$  позначає

нетрансльований і неповернений об'єкт  $T_i$ , у той час як  $M(\theta_i)$  є матрицею обертання.

Центр мас контейнера позначимо  $(x_0, y_0)$ . Нехай центр мас об'єкта  $T_i$  буде розміщений у  $v_i = (x_i, y_i)$  для  $i \in I_n$ . Позначимо систему об'єктів  $T_i$ ,  $i = 1, \dots, n$ , поміщених у контейнер  $\Omega$ , як  $\Omega_A$ .

Положення центра мас  $\Omega_A$  визначаємо так:

$$x_s(v) = \frac{1}{M} \sum_{i=0}^n w_i x_i, \quad y_s(v) = \frac{1}{M} \sum_{i=0}^n w_i y_i, \quad M = \sum_{i=0}^n w_i,$$

де  $v = (v_1, v_2, \dots, v_n)$ .

Нехай  $(\Delta x_e, \Delta y_e)$  – допустимі відхилення від заданої точки  $(x_e, y_e) \in \Omega$ .

Множина об'єктів  $T_i$ ,  $i = 1, \dots, n$ , розбивається на кластери  $\Lambda_1, \Lambda_2, \dots, \Lambda_N$  відповідно до заданих наборів індексів  $\Xi_1 = \{1, \dots, n_1\}$ ,  $\Xi_2 = \{n_1 + 1, \dots, n_2\}, \dots, \Xi_N = \{n_{N-1} + 1, \dots, n\}$ ,  $n = \sum_{q=1}^N m_q$ , де  $m_q = \text{card}(\Xi_q)$  для  $q \in \{1, \dots, N\} = J_N$ , тобто  $\Lambda_1 = \{T_1, \dots, T_{n_1}\}$ ,  $\Lambda_2 = \{T_{n_1+1}, \dots, T_{n_2}\}, \dots, \Lambda_N = \{T_{n_{N-1}+1}, \dots, T_n\}$ .

Вектори параметрів розміщення кластерів  $\Lambda_1, \Lambda_2, \dots, \Lambda_N$  позначають  $z_1 = (u_1, \dots, u_{n_1})$ ,  $z_2 = (u_{n_1+1}, \dots, u_{n_2}), \dots, z_N = (u_{n_{N-1}+1}, \dots, u_n)$ .

Далі вираз  $\Lambda_q(z_q)$  використовується для позначення кластера  $\Lambda_q$ , що містить об'єкти  $T_i(u_i)$ ,  $i \in \Xi_q$ , тобто  $\Lambda_q(z_q) = \bigcup_{i \in \Xi_q} T_i(u_i)$ .

Опукла оболонка об'єктів  $T_i(u_i)$ ,  $i \in \Xi_q$ , що визначається як  $\widehat{\Lambda}_q(z_q) = \text{conv} \Lambda_q(z_q)$ , пов'язана з кожним кластером  $\Lambda_q(z_q)$  для  $q \in J_N$ .

Відповідно до [45]: 1) два кластери  $\Lambda_q(z_q)$  і  $\Lambda_g(z_g)$  вважають такими, що не перетинаються, якщо їх опуклі оболонки не перетинаються, 2) кластер

$\Lambda_q(z_q)$  знаходиться у прямокутному контейнері  $\Omega$ , якщо його опукла оболонка знаходиться у контейнері  $\Omega$ .

Задача розрідженого компоунвання 3D кластерів за умови балансу (CLPB). Розмістити набір кластерів  $\Lambda_q(z_q)$ ,  $q \in J_N$ , що досить віддалені один від одного, у прямокутний контейнер  $\Omega$ , беручи до уваги умови балансу, так, що

$$\text{int } T_i(u_i) \cap \text{int } T_j(u_j) = \emptyset, \text{ для } i > j, (i, j) \in \Sigma_q \times \Sigma_q, q \in J_N, \quad (2.20)$$

$$\text{int } \hat{\Lambda}_q(z_q) \cap \text{int } \hat{\Lambda}_g(z_g) = \emptyset, \text{ для } q > g \in J_N, \quad (2.21)$$

$$\hat{\Lambda}_q(z_q) \subset \Omega \text{ для кожного } q \in J_N. \quad (2.22)$$

Умови балансу визначають так:

$$\mu(v) = \min\{\mu_1(v), \mu_2(v)\} \geq 0, \quad (2.23)$$

де

$$\begin{aligned} \mu_1(v) &= \min\{-(x_s(v) - x_e) + \Delta x_e, (x_s(v) - x_e) + \Delta x_e\}, \\ \mu_2(v) &= \min\{-(y_s(v) - y_e) + \Delta y_e, (y_s(v) - y_e) + \Delta y_e\}. \end{aligned}$$

Обмеження (2.20) описує умову неперетину об'єктів усередині одного кластера, обмеження (2.21) гарантує неперетин опуклих оболонок кластерів, обмеження (2.22) забезпечує розміщення кластерів у контейнері  $\Omega$ , а відношення (2.23) описує умови балансу.

**Математичне моделювання.** Математичну модель задачі дисперсного компонування кластерів у контейнері  $\Omega$  можна записати так:

$$\max_{(u, \phi, \psi, \rho) \in W \subset R^\sigma} \rho, \quad (2.24)$$

$$W = \{(u, \phi, \psi, \rho) \in R^\sigma : \Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}) - 0.5\rho \geq 0, \quad (2.25)$$

$$q > g \in I_N, \rho \geq 0,$$

$$\Phi^{T_i T_j}(u_i, u_j, u_{ij}) \geq 0, i > j, (i, j) \in \Xi_q \times \Xi_q, q \in I_N,$$

$$\Phi^{T_i \Omega^*}(u_i) \geq 0, i \in I_n, \mu(v) \geq 0\}.$$

Тут  $u = (u_1, u_2, \dots, u_n)$ ,  $u_i = (v_i, \theta_i)$ ,  $v_i = (x_i, y_i)$ ,  $\rho$  – відстань між  $\widehat{\Lambda}_q(z_q)$  і  $\widehat{\Lambda}_g(z_g)$ ,  $\psi = (\varphi_{qg}, \gamma_{qg}, q > g \in J_N)$  – допоміжні змінні у квазі-phi-функції  $\Phi^{\widehat{\Lambda}_q \widehat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$  для  $\widehat{\Lambda}_q(z_q)$ ,  $\widehat{\Lambda}_g(z_g)$ ,  $\phi = (u_{ij}, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$  є вектором допоміжних змінних у квазі-phi-функції  $\Phi^{T_i T_j}(u_i, u_j, u_{ij})$  для  $T_i(u_i)$  і  $T_j(u_j)$ ,  $\Phi^{T_i \Omega^*}(u_i)$  є phi-функцією для  $T_i(u_i)$  і  $\Omega^*$ . Явні вирази для (квазі) phi-функцій можна знайти у п. 2.3. Детальну інформацію щодо методу phi-функцій наведено, наприклад, у [228–230]. Тут  $\sigma = 1 + 3n + \frac{\tau_q}{2} \sum_{q=1}^N m_q(m_q - 1) + N(N - 1)$  – кількість змінних,  $\tau_q$  – кількість допоміжних змінних,  $(i, j) \in \Xi_q \times \Xi_q, q \in I_N$  а  $\mu(v) \geq 0$  – умова балансу, визначена у (2.23).

Задача нелінійного програмування (2.24), (2.25) містить усі оптимальні розв'язки оптимізаційної задачі розрідженого компонування 3D кластерів за умови балансу.

Задача (2.24), (2.25) має  $O(\sum_{q=1}^N m_q^2) + O(N^2)$  нелінійних нерівностей та

$O(\sum_{q=1}^N m_q^2) + O(N^2)$  змінних, враховуючи допоміжні змінні у квазі-phi-

функціях.

Далі описано алгоритм знаходження локального-оптимального розв'язку задачі (2.24), (2.25).

**Алгоритм розв'язання задачі.** Використовується стратегія розв'язання, описана в п. 2.5. Застосовується багатостартова стратегія (multistart), що має такі етапи:

Етап 1. Побудова набору допустимих стартових точок задачі (2.24), (2.25).

Етап 2. Знаходження набору локальних максимумів для (2.24), (2.25) з використанням стартової точки, отриманої на етапі 1.

Етап 3. Вибір найкращих локальних максимумів, знайдених на етапі 2.

Запропоновано алгоритм ECSP для побудови допустимих стартових точок задачі (2.24), (2.25). Цей алгоритм передбачає такі кроки.

Крок 1. Для кожного кластера  $\Lambda_q(z_q)$  будується окіл  $\lambda C_q$  із змінним центром  $v_q = (x_q, y_q)$  і змінним радіусом  $\lambda R_q$ , де  $\lambda$  є параметром масштабування,  $R_q = \sqrt{\sum_{i \in \Xi_q} S_i}$ . Тут  $S_i$  – площа основи об'єкта  $T_i$ ,  $i \in \Xi_q$ ,  $q \in I_N$ .

Крок 2. Розв'язання задачі нелінійного програмування на знаходження змінних центрів  $v_q = (x_q, y_q)$  відповідних околів  $\lambda C_q$  для  $\lambda = 0$ ,  $q \in I_N$ , з

урахуванням умови балансу (2.23), де  $x_s(v) = \frac{1}{M} \sum_{q=0}^N w_q^c x_q$ ,

$$y_s(v) = \frac{1}{M} \sum_{q=0}^N w_q^c y_q, \quad w_0^c = w_0, \quad w_q^c = \sum_{i=1}^{n_q} w_i - \text{вага кластера, } q \in I_N:$$

$$\min_{(v, \xi) \in V} \xi, \quad (2.26)$$

$$V = \{(v, \xi) \in R^{2N+1} : \varphi_q(v_q) \geq 0, q \in I_N, \xi - \mu^c(v) \geq 0\}, \quad (2.27)$$

$$\mu^c(v) = \min\{\mu_1^c(v), \mu_2^c(v)\}, \quad (2.28)$$

$$\mu_1^c(v) = \min\{-(x_s^c(v) - x_e) + \Delta x_e, (x_s^c(v) - x_e) + \Delta x_e\},$$

$$\mu_2^c(v) = \min\{-(y_s^c(v) - y_e) + \Delta y_e, (y_s^c(v) - y_e) + \Delta y_e\},$$

$$\varphi_q(v_q) = \min_{s=1, \dots, 4} f_{is}(v_i),$$

$$f_{i1}(v_i) = x_i, f_{i2}(v_i) = y_i, f_{i3}(v_i) = -x_i + l, f_{i4}(v_i) = -y_i + w.$$

Початкові значення  $v_q = (x_q, y_q) \in \Omega$  для розв'язання задачі (2.26), (2.27) генеруються випадково. Позначимо  $(v^0, \xi^0)$  локальний розв'язок задачі (2.26), (2.27), де  $v^0 = (v_1^0, \dots, v_N^0)$ .

Крок 3. Починаючи з точки  $(v^0, \lambda^0 = 0)$ , розв'язуємо таку задачу:

$$\max_{(v, \lambda) \in V} \lambda, \quad (2.29)$$

$$V = \{(v, \lambda) \in R^{2N+1} : \Phi^{C_q C_g}(v_q, v_g, \lambda) \geq 0, q > g \in I_N\}, \quad (2.30)$$

$$\Phi^{C_q \Omega^*}(v_q, \lambda) \geq 0, q \in I_N, \lambda \geq 0, \mu^c(v) \geq 0\},$$

де  $\Phi^{C_q C_g}(v_q, v_g, \lambda)$  є phi-функцією для  $\lambda C_q$  і  $\lambda C_g$  з відповідними радіусами  $\lambda R_q$  і  $\lambda R_g$  та центрами  $v_q = (x_q, y_q)$  і  $v_g = (x_g, y_g)$  для  $q > g \in I_N$ . Тут  $\Phi^{C_q \Omega^*}(v_q, \lambda)$  є phi-функцією для кола  $\lambda C_q$  з центром у  $v_q = (x_q, y_q)$  і об'єкта  $\Omega^* = R^2 \setminus \text{int } \Omega, q \in I_N$ . Функцію  $\mu^c(v)$  визначено в (2.28). Локальний розв'язок (2.29), (2.30) позначаємо  $(v^*, \lambda^*)$ .

Крок 4. Для кожного околу  $\lambda^* C_q$  генеруємо випадкові вектори трансляції  $v_i = (x_i, y_i) \in \lambda^* C_q$  об'єктів  $T_i(v_i), i \in \Xi_q$ , з кластера  $\Lambda_q(z_q), q \in I_N$ .

Крок 5. Для кожного кластера  $\Lambda_q, q \in I_N$ , формуємо вектор  $z_q = (v_{n_{q-1}+1}, \dots, v_{n_q})$  і знаходимо мінімальне відхилення центра мас  $(x_s^q(z^q), y_s^q(z^q))$  кластера від точки  $v_q^0$  за допомогою розв'язання такої задачі нелінійного програмування:

$$\min_{(z^q, \xi^q) \in V^q} \xi^q, \quad (2.31)$$

$$V^q = \{(z^q, \xi^q) \in R^{2n_q+1} : \varphi_i(v_i) \geq 0, i \in \Xi_q, \xi^q - \mu^q(z^q) \geq 0\}, \quad (2.32)$$

$$\mu^q(z^q) = \min\{\mu_1^q(z^q), \mu_1^q(z^q)\}, \quad (2.33)$$

$$\mu_1^q(z^q) = \min\{-(x_s^q(z^q) - x_q^0) + \Delta x_e, (x_s^C(z^q) - x_q^0) + \Delta x_e\},$$

$$\mu_2^q(z^q) = \min\{-(y_s^q(z^q) - y_q^0) + \Delta y_e, (y_s^C(z^q) - y_q^0) + \Delta y_e\},$$

$$\varphi_i^q(v_i^q) = (\lambda^* R^q)^2 - (x_q^* - x_i)^2 - (y_q^* - y_i)^2,$$

$$\text{де } x_s^q(z^q) = \frac{1}{M_q} \sum_{i \in \Xi_q} w_i x_i, \quad y_s^q(z^q) = \frac{1}{M_q} \sum_{i \in \Xi_q} w_i y_i, \quad M_q = \sum_{i \in \Xi_q} w_i.$$

Нехай локальний максимум для задачі (2.31), (2.33) становитиме  $(z^{q*}, \xi^{q*})$ .

Крок 6. Початкові значення  $\varphi_{qg}^0, \gamma_{qg}^0$  отримують нескладними геометричними розрахунками для квазі-phi-функції  $\Phi^{\hat{\Lambda}_q \hat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg})$ ,  $q > g \in I_N$  у задачі (2.24), (2.25). Ці значення використовуються для визначення відокремлювальних прямих між околами  $\lambda^* C_q$  і  $\lambda^* C_g$  радіусами  $\lambda^* R_q$  і  $\lambda^* R_g$  з центральними точками  $v_q^*$  і  $v_g^*$ ,  $q > g \in I_N$ .

Крок 7. Допустимі стартові точки для задачі (2.24), (2.25) отримують за допомогою розв'язання такої задачі нелінійного програмування:

$$\max_{(u, \phi, \psi, \beta) \in G} \beta, \quad (2.34)$$

$$G = \{(u, \phi, \psi, \beta) \in R^\sigma : \Phi^{\hat{\Lambda}_q \hat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}, \beta) \geq 0, \quad (2.35)$$

$$q > g \in I_N, 0 \leq \beta \leq 1,$$

$$\Phi^{T_i T_j}(u_i, u_j, u_{ij}, \beta) \geq 0, i > j, (i, j) \in \Xi_q \times \Xi_q, q \in I_N,$$

$$\Phi^{T_i \Omega^*}(u_i, \beta) \geq 0, i \in I_n, \mu(v) \geq 0 \},$$

де  $u = (u_1, u_2, \dots, u_n)$ ,  $u_i = (v_i, \theta_i)$ ,  $v_i = (x_i, y_i)$ ,  $\psi = (\varphi_{qg}, \gamma_{qg}, q > g \in J_N)$ ;  $\beta$  – коефіцієнт масштабування об'єктів;  $\phi = (u_{ij}, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$  позначає допоміжні змінні у квазі-phi-функції для масштабованих об'єктів  $\beta T_i(u_i)$  і  $\beta T_j(u_j)$ ;  $\Phi^{T_i T_j}(u_i, u_j, u_{ij}, \beta)$  є квазі-phi-функцією для  $\beta T_i(u_i)$  і  $\beta T_j(u_j)$ ;

$\Phi^{T_i \Omega^*}(u_i, \beta)$  є phi-функцією для  $\beta T_i(u_i)$  і  $\Omega^*$ ;  $\Phi^{\hat{\Lambda}_q \hat{\Lambda}_g}(z_q, z_g, \varphi_{qg}, \gamma_{qg}, \beta)$  є квазі-phi-функцією для двох кластерів;  $\sigma = 1 + 3n + \frac{\tau_q}{2} \sum_{q=1}^N m_q(m_q - 1) + N(N - 1)$  – кількість змінних задачі,  $\tau_q$  – кількість допоміжних змінних.

Допустима стартова точка  $(u^0, \phi^0, \psi^0, \beta^0 = 0)$  використовується для розв'язання задачі (2.34), (2.35). Тут  $\psi^0 = (\varphi_{qg}^0, \gamma_{qg}^0, q > g \in J_N)$  отримують на етапі 6;  $v_i^0 = (x_i^0, y_i^0) \in C_q$  знайдено на кроці 5 та  $\theta_i^0 \in [0, 2\pi]$  є випадково згенерованим параметром обертання  $\beta T_i(u_i)$  для  $i \in \Xi_q, q \in I_N$ ;  $\phi^0 = (u_{ij}^0, (i, j) \in \Xi_q \times \Xi_q, q \in I_N)$ , вектор  $u_{ij}^0$  допоміжних змінних отримують простим геометричним предметом  $\Phi^{T_i T_j}(u_i^0, u_j^0, u_{ij}^0, \beta^0 = 0) \geq 0$ .

Глобальний розв'язок задачі (2.34), (2.35),  $(\beta^* = 1)$ , дає розв'язок вихідної задачі (2.24), (2.25). Його можна використовувати як вихідну точку для задачі (2.24), (2.25).

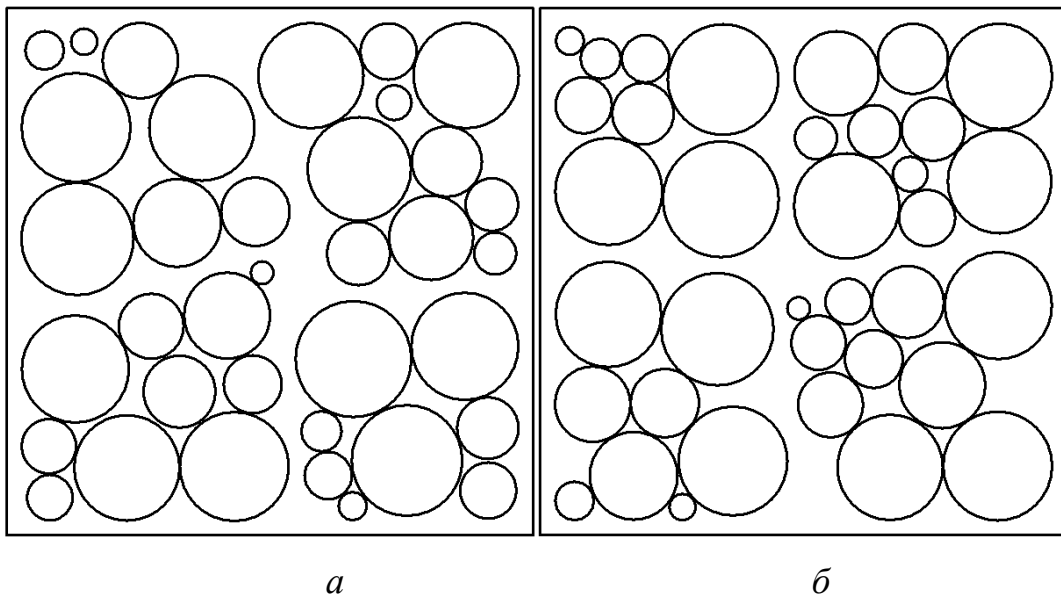
**Приклади компонування кластерів у прямокутному контейнері за умови балансу.** Щоб показати ефективність запропонованого підходу до дисперсного компонування 3D кластерів за умови балансу, використано приклади з п. 2.6. Для розв'язання задач нелінійного програмування застосовано IPOPT <https://projects.coin-or.org/Ipopt> [231].

Для задачі балансного компонування 3D кластерів у контейнер  $\Omega$  розглянуто шість прикладів з п. 2.6, де  $l=15, w=15, w_0=0, (x_e; y_e) = (7,5; 7,5)$  для прикладів 2.9–2.14 і  $(x_e; y_e) = (6,5; 6,5)$  для прикладу 2.15,  $\Delta x_e = \Delta y_e = 0,05$ . Отримано обчислювальні результати для компонування 3D кластерів із умовами балансу та без них. Програму було запущено по 100 стартів для кожного прикладу.

Приклад 2.9. Наведено набір  $n=36$  кіл із прикладу 2 [45]. Кола розділені на кластери  $\Lambda_q(z_q)$ ,  $q=1, 2, 3, 4$ :  $m_1=10$ ,  $m_2=10$ ,  $m_3=8$ ,  $m_4=8$ ;

$\{w_i, i=1, \dots, 36\} = \{2.235, 2.250, 2.253, 1.454, 1.018, 0.832, 0.664, 0.585, 0.386, 0.275, 0.113, 2.403, 2.350, 2.286, 1.535, 1.086, 0.893, 0.719, 0.628, 0.428, 0.310, 0.144, 2.576, 2.452, 2.320, 1.618, 1.156, 0.956, 0.776, 0.672, 0.472, 0.348, 0.180, 2.756, 2.557, 2.353\}$ .

Отриманий результат: а) без умов балансу  $\rho^* = 0,410928$ , час процесора – 137,796 с; б) з умовами балансу  $\rho^* = 0,424021$ ,  $\mu_1(v^*) = 0,008224$ ,  $\mu_2(v^*) = 0,049992$ , час процесора – 275,919 с. Відповідні локальні оптимальні компоунвання наведено на рис. 2.9.

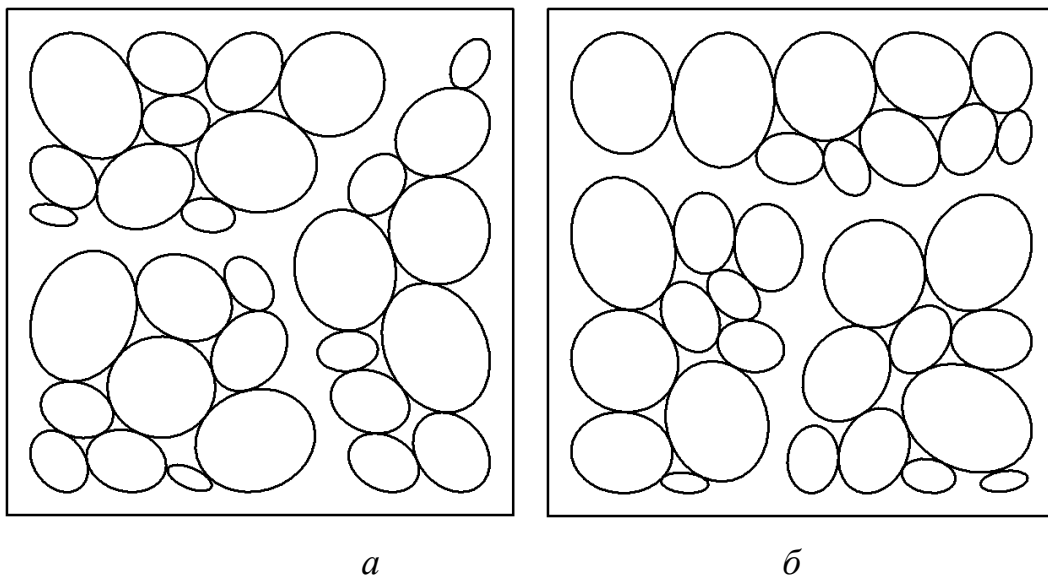


**Рис. 2.9.** Оптимізовані компоунвання кластерів для прикладу 2.9: *a* – без умов балансу; *б* – з умовами балансу

Приклад 2.10. Наведено набір  $n=30$  еліпсів із прикладу 1 у [45]. Сукупність еліпсів розділена на кластери  $\Lambda_q(z_q)$ ,  $q=1, 2, 3$ :  $m_1=10$ ,  $m_2=10$ ,  $m_3=10$ ;

$\{w_i, i=1, \dots, 30\} = (3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750, 0.540, 0.400, 0.210, 3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750, 0.540, 0.400, 0.210, 3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750)$ .

Отриманий результат: а) без умов балансу  $\rho^* = 0,691521$ , час процесора – 582,586 с; б) з умовами балансу  $\rho^* = 0,688519$ ,  $\mu_1(v^*) = 0,019491$ ,  $\mu_2(v^*) = 0,037263$ , час процесора – 1103,894 с. Відповідні локальні оптимальні компонування наведено на рис. 2.10

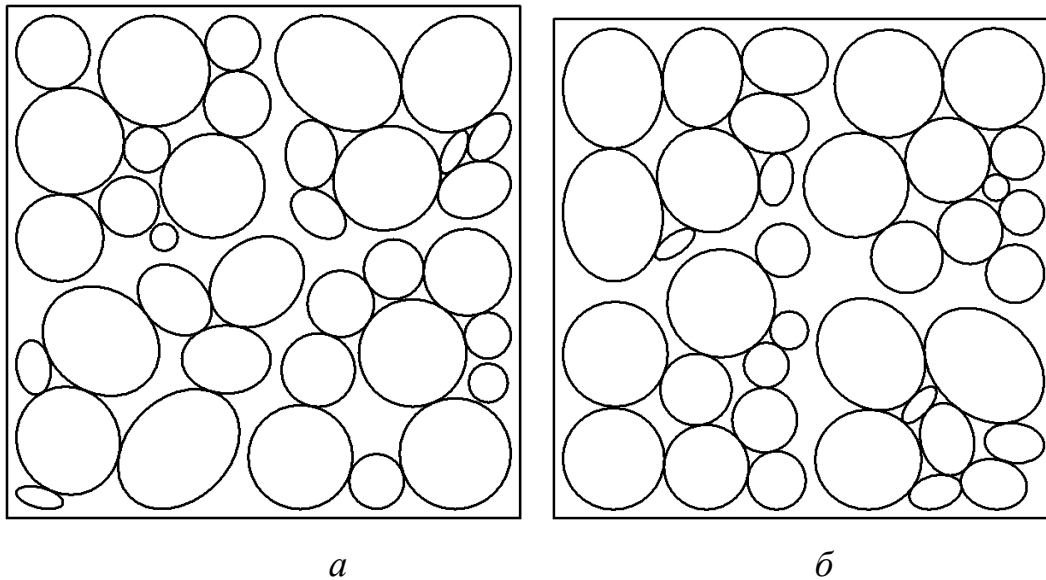


**Рис. 2.10.** Оптимізовані компонування кластерів для прикладу 2.10: *а* – без умов балансу; *б* – з умовами балансу.

Приклад 2.11. Наведено набір  $n=36$  об'єктів з прикладу 4 [45]. Об'єкти розділені на кластери  $\Lambda_q(z_q)$ ,  $q=1, 2, 3, 4$ ;  $m_1=10$  кола,  $m_2=10$  кола,  $m_3=8$  еліпси,  $m_4=8$  еліпси;

$\{w_i, i=1, \dots, 36\} = (2.641, 2.481, 2.326, 1.626, 1.156, 0.951, 0.766, 0.660, 0.456, 0.331, 0.160, 2.641, 2.481, 2.326, 1.626, 1.156, 0.951, 0.766, 0.660, 0.456, 0.400, 0.210, 3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750, 0.540, 0.400, 0.210, 3.000, 2.700, 2.400)$ .

Отриманий результат: а) без умов балансу  $\rho^* = 0,272577$ ,  $\mu_1(v^*) = 0,008624$ ,  $\mu_2(v^*) = 0,049986$ , час процесора – 317,026 с; б) з умовами балансу  $\rho^* = 0,269150$ , час процесора – 384,371с. Відповідні локальні оптимальні компоновки наведено на рис. 2.11



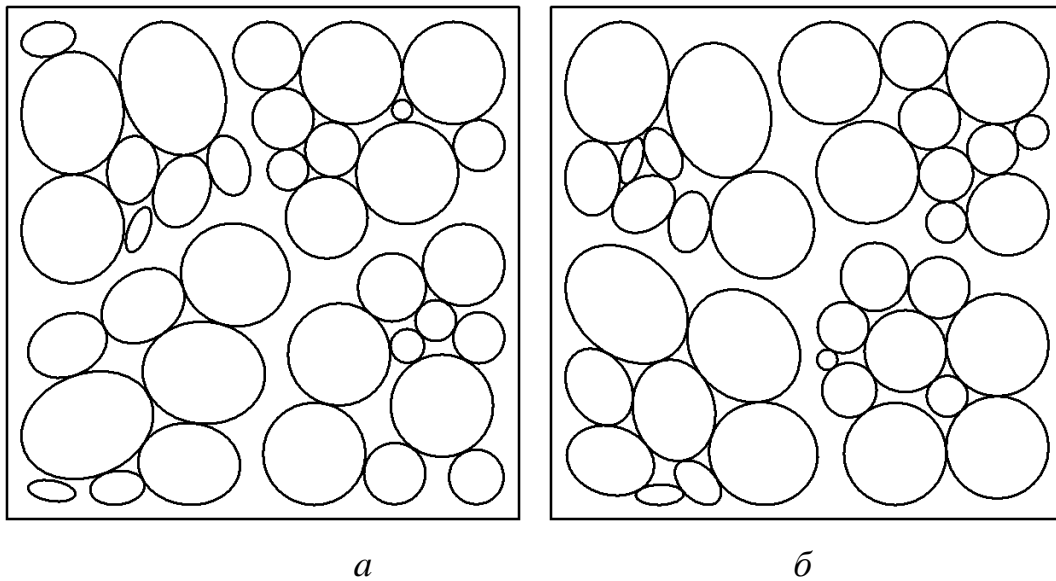
**Рис. 2.11.** Оптимізовані компоновки кластерів для прикладу 2.11: *а* – без умов балансу; *б* – з умовами балансу

Приклад 2.12. Наведено набір  $n=36$  об'єктів з прикладу 3 [45]. Еліпси розділені на кластери  $\Lambda_q(z_q)$ ,  $q=1, 2, 3, 4$ :  $m_1=10$  кола,  $m_2=10$  кола,  $m_3=8$  еліпси,  $m_4=8$  еліпси;

$\{w_i, i=1, \dots, 36\} = \{2.250, 2.250, 2.250, 1.440, 1.000, 0.810, 0.640, 0.563, 0.360, 0.250, 0.090, 2.250, 2.250, 2.250, 1.440, 1.000, 0.810, 0.640, 0.563, 0.360, 0.400, 0.210, 3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750, 0.540, 0.400, 0.210, 3.000, 2.700, 2.400\}$ .

Отриманий результат: а) без балансу умов  $\rho^* = 0,422594$ , час процесора – 312,626 с; б) з умовами балансу  $\rho^* = 0,421740$ ,  $\mu_1(v^*) = 0,049982$ ,

$\mu_2(v^*) = 0,049982$ , час процесора – 1925,068 с. Відповідні локальні оптимальні компонування наведено на рис. 2.12.

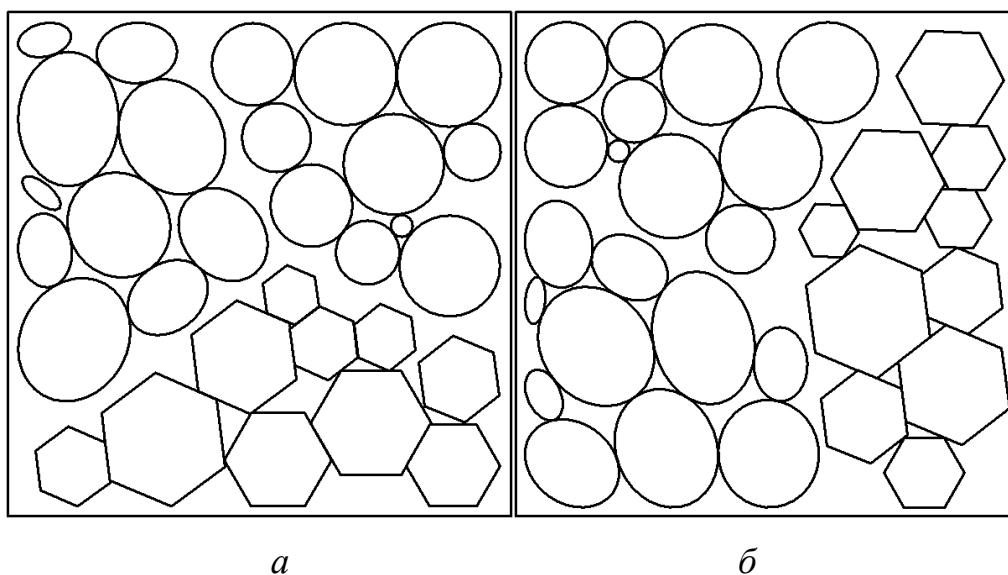


**Рис. 2.12.** Оптимізовані компонування кластерів для прикладу 2.12: *a* – без умов балансу; *б* – з умовами балансу

Приклад 2.13. Наведено набір  $n=30$  об'єктів з прикладу 6 [45]. Об'єкти розділені на кластери  $\Lambda_q(z_q)$ ,  $q=1,2,3$ :  $m_1=10$  опуклі багатокутники,  $m_2=10$  кола,  $m_3=10$  еліпси;

$\{w_i, i=1, \dots, 30\} = (4.000, 3.240, 2.890, 2.560, 1.960, 1.690, 1.440, 1.210, 1.000, 0.810, 0.113, 2.403, 2.350, 2.323, 2.289, 1.543, 1.513, 1.098, 0.904, 0.729, 0.400, 0.210, 3.000, 3.040, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880)$ .

Отриманий результат: а) без умов балансу  $\rho^* = 0,304910$ , час процесора – 1564,550 с; б) з умовами балансу  $\rho^* = 0,269486$ ,  $\mu_1(v^*) = 0,050002$ ,  $\mu_2(v^*) = 0,032384$ , час процесора – 1990,042 с. Відповідні локальні оптимальні компонування наведено на рис. 2.13.

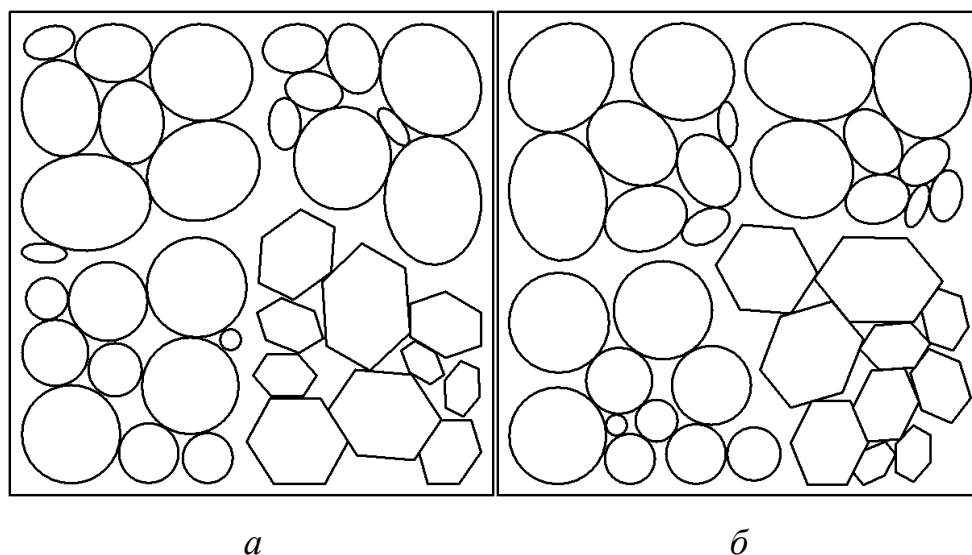


**Рис. 2.13.** Оптимізовані компонування кластерів для прикладу 2.13: *a* – без умов балансу; *б* – з умовами балансу

Приклад 2.14. Наведено набір  $n = 36$  об'єктів з прикладу 5 [45]. Об'єкти розділені на кластери  $\Lambda_q(z_q)$ ,  $q = 1, 2, 3, 4$ :  $m_1 = 10$  опуклі багатокутники,  $m_2 = 10$  кола,  $m_3 = 8$  еліпси,  $m_4 = 8$  еліпси;

$\{w_i, i = 1, \dots, 36\} = (3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750, 0.540, 0.400, 0.113, 2.403, 2.350, 2.286, 1.535, 1.086, 0.893, 0.719, 0.628, 0.428, 0.400, 0.210, 3.000, 2.700, 2.400, 1.800, 1.300, 1.080, 0.880, 0.750, 0.540, 0.400, 0.210, 3.000, 2.700, 2.400)$ .

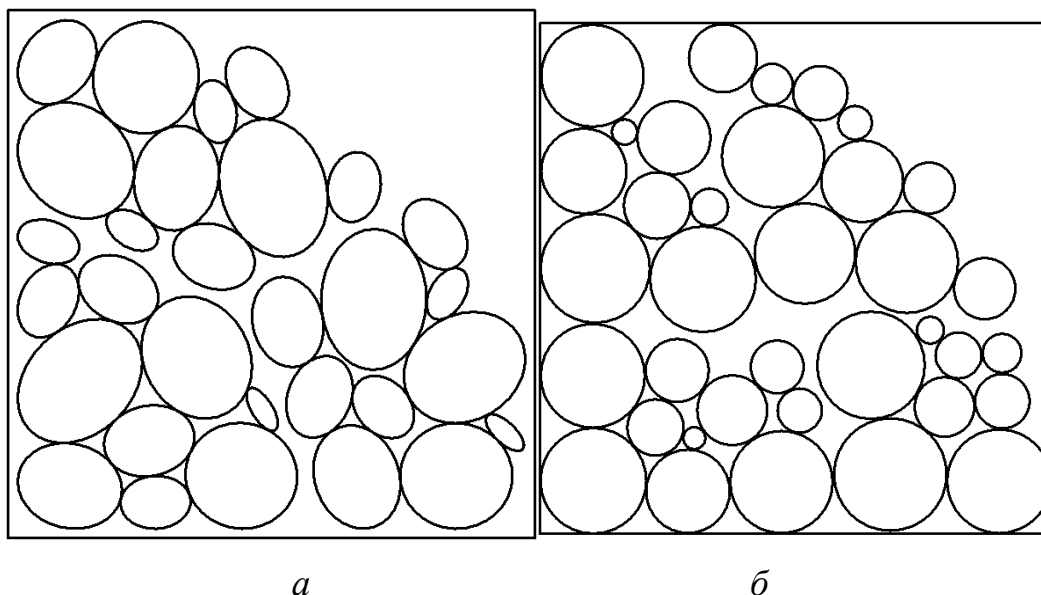
Отриманий результат: а) без умов балансу  $\rho^* = 0,370691$ , час процесора – 1543,848 с; б) з умовами балансу  $\rho^* = 0,344749$ ,  $\mu_1(v^*) = 0,050010$ ,  $\mu_2(v^*) = 0,049998$ , час процесора – 1503,132 с. Відповідні локальні оптимальні компонування наведено на рис. 2.14.



**Рис. 2.14.** Оптимізовані компонування кластерів для прикладу 2.14: *a* – без умов балансу; *б* – з умовами балансу

Приклад 2.15. Набір  $n = 30$  еліпсів з прикладу 1 [45] подано з  $(x_e; y_e) = (6,5; 6,5)$ . Отриманий результат: з умовами балансу –  $\rho^* = 0,273300$ ,  $\mu_1(v^*) = 0,050025$ ,  $\mu_2(v^*) = 0,050028$ . Час процесора становить 542,01 с. Відповідне локальне оптимальне компонування наведено на рис. 2.15, *a*.

Набір з  $n = 36$  кіл з прикладу 2 [45] подано з  $(x_e; y_e) = (6,5; 6,5)$ . Отриманий результат: з умовами балансу –  $\rho^* = 0,014361$ ,  $\mu_1(v^*) = 0,049988$ ,  $\mu_2(v^*) = 0,050003$ . Час процесора – 355,183 с. Відповідне локальне оптимальне компонування наведено на рис. 2.15, *б*.



**Рис. 2.15.** Оптимізовані компонування кластерів з умовами балансу для прикладу 2.15: *a* – для еліпсів; *б* – для кіл

Приклад 2.16. Задано набір з  $n = 18$  складених об'єктів, який поділено на  $N=4$  кластери  $\Lambda_q(\eta_q)$ ,  $q=1, \dots, 4$ ,  $m_1 = 5$ ,  $m_2 = 5$ ,  $m_3 = 4$ ,  $m_4 = 4$ . . Контейнер – пряма призма з багатокутною основою, заданою вершинами:

$\{(x_k, y_k), k = 1, \dots, 9\} = \{(-7.266244, 1.593456), (-5.941336, -6.880334), (-3.291530, -8.733971), (2.310916, -10.663272), (6.702019, -10.663272), (16.052038, -4.080957), (20.670273, 2.387873), (7.118418, 9.915923), (-1.853066, 7.948795)\}$ .

Кластер  $\Lambda_1$  містить складені об'єкти  $\mathbf{T}_i(u_i), i=1, \dots, 5$ , кожний компонент яких є прямою призмою з багатокутною основою.

Компонент 1.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(2.000000, 0.000000), (1.000000, 1.299038), (-1.000000, 1.299038), (-2.000000, -0.000000), (-1.000000, -1.299038), (1.000000, -1.299038)\}$ .

Компонент 2.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(-0.250000, 0.000000), (-1.250000, 1.299038), (-3.250000, 1.299038), (-4.250000, -0.000000), (3.250000, -1.299038), (-1.250000, -1.299038)\}$ .

Компонент 1.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(-1.600000, 0.000000), (0.800000, 1.299038), (-0.800000, 1.299038), (-1.600000, -0.000000), (-0.800000, -1.299038), (0.800000, -1.299038)\}$ .

Компонент 2.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(-0.425000, 0.000000), (-1.225000, 1.299038), (-2.825000, 1.299038), (-3.625000, -0.000000), (-2.825000, -1.299038), (-1.225000, -1.299038)\}$ .

Компонент 1.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(1.300000, 0.000000), (0.650000, 0.866025), (-0.650000, 0.866025), (-1.300000, -0.000000), (-0.650000, -0.866025), (0.650000, -0.866025)\}$ .

Компонент 2.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(0.125000, 0.000000), (-0.775000, 0.866025), (-2.075000, 0.866025), (-2.725000, -0.000000), (-2.075000, -0.866025), (-0.775000, -0.866025)\}$ .

Компонент 1.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(1.100000, 0.000000), (0.550000, 0.692820), (-0.550000, 0.692820), (-1.100000, -0.000000), (-0.550000, -0.692820), (0.550000, -0.692820)\}$ .

Компонент 2.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(-0.062500, 0.000000), (-0.612500, 0.692820), (-1.712500, 0.692820), (-2.262500, -0.000000), (-1.712500, -0.692820), (-0.612500, -0.692820)\}$ .

Компонент 1.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(0.900000, 0.000000), (0.450000, 0.519615), (-0.450000, 0.519615), (-0.900000, -0.000000), (-0.450000, -0.519615), (0.450000, -0.519615)\}$ .

Компонент 2.  $\{(x_k, y_k), k = 1, \dots, 6\} = \{(0.075000, 0.000000), (-0.375000, 0.519615), (-1.275000, 0.519615), (-1.725000, -0.000000), (-1.275000, -0.519615), (-0.375000, -0.519615)\}$ .

Кластер  $\Lambda_2$  містить складені об'єкти  $\mathbf{T}_i(u_i), i = 6, \dots, 10$ , в яких компонент 1 є циліндром з коловою основою, а компонент 2 – циліндр з еліптичною основою.

Компонент 1.  $r = 0.300000, x_c = 0, y_c = 0, \theta_c = 0$ .

Компонент 2.  $a=2, b=1.5, x_c = -1.35, y_c = 0, \theta_c = 0.$

Компонент 1.  $r=1.5, x_c = -1.35, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=1.6, b=1.5, x_c = -2.25, y_c = 0, \theta_c = 0.$

Компонент 1.  $r=1.2, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=1.3, b=1, x_c = -1.65, y_c = 0, \theta_c = 0.$

Компонент 1.  $r=0.9, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=1.1, b=0.8, x_c = -1.27, y_c = 0, \theta_c = 0.$

Компонент 1.  $r=0.75, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=0.9, b=0.8, x_c = -1.0125, y_c = 0, \theta_c = 0.$

Кластер  $\Lambda_3$  містить складені об'єкти  $\mathbf{T}_i(u_i), i=11, \dots, 14$ , кожний компонент яких є циліндром з еліптичною основою.

Компонент 1.  $a=0.8, b=0.5, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=2, b=1.5, x_c = -0.6, y_c = 0, \theta_c = 0.$

Компонент 1.  $a=2, b=1.5, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=1.6, b=1.5, x_c = -2.25, y_c = 0, \theta_c = 0.$

Компонент 1.  $a=1.6, b=1.5, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=1.3, b=1, x_c = -2.025, y_c = 0, \theta_c = 0.$

Компонент 1.  $a=1.3, b=1, x_c = 0, y_c = 0, \theta_c = 0.$

Компонент 2.  $a=1.1, b=0.8, x_c = -1.425, y_c = 0, \theta_c = 0.$

Кластер  $\Lambda_4$  містить складені об'єкти  $\mathbf{T}_i(u_i), i=15, \dots, 18$ , кожний компонент яких є циліндром з еліптичною основою.

Компонент 1.  $a=1.1, b=0.8, x_c = 0, y_c = 0, \theta_c = 0$ .

Компонент 2.  $a=2, b=1.5, x_c = -1.1625, y_c = 0, \theta_c = 0$ .

Компонент 1.  $a=0.9, b=0.6, x_c = 0, y_c = 0, \theta_c = 0$ .

Компонент 2.  $a=1.6, b=1.5, x_c = -0.825, y_c = 0, \theta_c = 0$ .

Компонент 1.  $a=0.7, b=0.3, x_c = 0, y_c = 0, \theta_c = 0$ .

Компонент 2.  $a=1.3, b=1, x_c = -1.35, y_c = 0, \theta_c = 0$ .

Компонент 1.  $a=1.8, b=1.5, x_c = 0, y_c = 0, \theta_c = 0$ .

Компонент 2.  $a=1.1, b=0.8, x_c = -2.25, y_c = 0, \theta_c = 0$ .

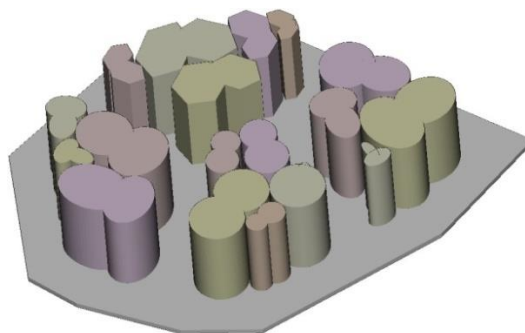
Результат розв'язання:

Параметри розміщення складених об'єктів, що відповідають точці локального максимуму:

$\{(x_i, y_i, \theta_i), i = 1, \dots, 18\} = \{(4.636438, -2.656283, 4.712389), (7.376709, -3.593916, 4.600817), (5.888803, -7.141026, 6.045830), (6.171510, -0.466975, 4.213783), (9.279648, -5.209521, 1.302087), (16.233805, 1.348533, 5.949953), (10.320549, 1.847104, 3.864044), (12.538577, -2.288103, 2.558323), (12.011259, 3.550484, 0.432171), (13.111199, 2.320593, 2.668424), (-3.356138, -4.068801, 4.449639), (-0.574621, -3.291340, 1.589801), (-3.243807, 0.405375, -1.583050), (-0.233280, -6.307278, 0.261432), (2.749412, 2.902154, 2.947840), (5.695418, 3.315449, 2.565929), (-0.654763, 4.652653, 3.188920), (6.362203, 5.847768, 6.077668)\}.$

Значення цільової функції становить  $\rho^* = 2,288532$ , час обчислення дорівнює 1191,48 секунд.

Відповідне оптимізоване розріджене компоновання наведено на рис. 2.16.



**Рис. 2.16.** Оптимізоване розріджене компоновання кластерів для прикладу 2.16

Приклад 2.17. Задано набір з  $n = 18$  складених об'єктів, який поділено на  $N=4$  кластери  $\Lambda_q(\eta_q)$ ,  $q = 1, \dots, 4$ ,  $m_1 = 5$ ,  $m_2 = 5$ ,  $m_3 = 4$ ,  $m_4 = 4$ . Контейнер – кубоїд з прямокутною основою розмірами  $15 \times 15 \times 15$ .

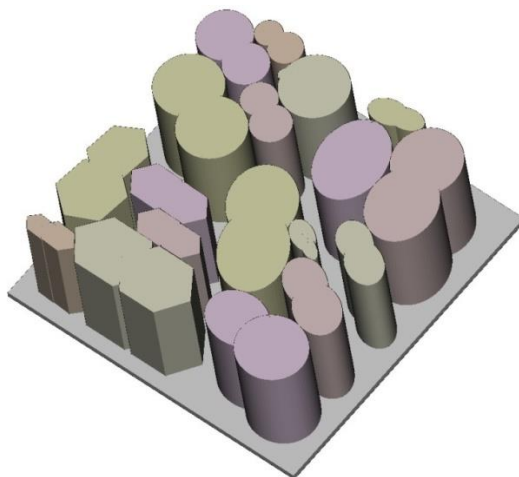
Результат розв'язання:

Параметри розміщення складених об'єктів, що відповідають точці локального максимуму:

$$\{(x_i, y_i, \theta_i), i = 1, \dots, 18\} = \{(4.607481 \ 1.953990 \ 3.141593), (2.047233 \ 3.726344 \ 1.439999), (4.090306 \ 5.390742 \ 3.114465), (5.370780 \ 3.945848 \ 3.141593), (2.283477 \ 1.217252 \ -0.069521), (5.057482 \ 12.583052 \ 2.946282), (2.154952 \ 9.903465 \ 3.604256), (1.854952 \ 12.618914 \ 3.483375), (6.024508 \ 10.451653 \ 0.511282), (4.260494 \ 13.595048 \ 0.148695), (10.417896 \ 7.064433 \ 3.633114), (9.215142 \ 5.350219 \ 1.056966), (12.746897 \ 2.156921 \ 0.138528), (13.115414 \ 4.669881 \ 0.532753), (13.316911 \ 7.487426 \ 0.544487), (9.618554 \ 13.664098 \ 2.939621), (9.621533 \ 12.356390 \ 4.572455), (12.831322 \ 10.546613 \ 1.364929)\}.$$

Значення цільової функції становить  $\rho^* = 0,654953$ , час обчислення дорівнює 1458,56 секунд.

Відповідне оптимізоване розріджене компонування наведено на рис. 2.17.



**Рис. 2.17.** Оптимізоване розріджене компонування кластерів для прикладу 2.17

Деякі питання, аналіз яких наведено в розділі, розглянуто в працях [45, 47, 48, 58].

### РОЗДІЛ 3. ОПТИМІЗАЦІЙНІ ЗАДАЧІ БАЛАНСНОГО КОМПОНУВАННЯ ВАНТАЖІВ У КОНТЕЙНЕРАХ

У розділі побудовано математичні моделі низки оптимізаційних задач рівноважного компонування контейнерів, зокрема, з урахуванням комбінаторних особливостей задач, спеціальних геометричних та балансних умов, багатокритеріальності, а також запропоновано стратегії їх розв'язання.

#### 3.1. Комбінаторні конфігурації в оптимізаційних задачах балансного компонування контейнерів

Задачі балансного компонування належать до класу NP-складних задач розміщення [117] і є предметом досліджень обчислювальної геометрії [87], а методи їх розв'язання – новим напрямом теорії дослідження операцій [95]. Суть задачі полягає в пошуку оптимального розміщення заданого набору 3D-об'єктів у деякій обмеженій зоні (контейнері) з урахуванням обмежень, які забезпечують баланс такої системи.

Необхідність врахування обмежень балансу в оптимізаційних задачах розміщення виникає в різних прикладних галузях науки і техніки, наприклад, у задачах логістики (у разі пакування вантажів для транспортування або зберігання), в машинобудуванні (у разі компонування літальних апаратів, судів, підводних човнів, обладнання, приладів та пристроїв, деталей виробів). Особливу увагу цей клас задач привертає під час проектування ракетно-космічної техніки. На початковому етапі проектування (компонування) космічного апарата необхідно враховувати низку обмежень на статичні і динамічні характеристики (центр мас, осьові і відцентрові моменти інерції) [118].

У працях [105, 119–122] розглянуто задачі компонування циліндрів у циліндричному контейнері з обмеженнями поведінки, врахування яких необхідне для дотримання зрівноважування супутникової системи. У цих

публікаціях наведено математичні моделі з різними функціями цілі. Для розв'язання таких задач запропоновано евристичні алгоритми, що враховують особливості кожної задачі.

Математичні моделі низки оптимізаційних задач балансного компонування можуть бути побудовані з урахуванням особливостей їх дискретної структури. Для математичного моделювання і розв'язання таких задач потрібно використовувати комбінаторні конфігурації, що мають відповідні властивості. Основні підходи до математичного моделювання та розв'язання задач оптимізації на комбінаторних конфігураціях описано в монографіях [1, 10, 111].

У працях [123, 124] наведено математичні моделі і методи розв'язання задач балансного компонування (Balance Layout Problem, BLP) множини 3D-об'єктів у заданому контейнері. Контейнер розділений круговими стелажми на підконтейнери. При цьому розбиття множини об'єктів на підмножини відповідно до розміщення об'єктів усередині підконтейнерів вважають заданим.

Розглянемо задачу балансного компонування 3D-об'єктів (куля, циліндр, тор, сфероциліндр, паралелепіпед, правильна призма) в контейнері (форми паралелепіпеда, циліндра, параболоїда обертання або зрізаного конуса), розділеному горизонтальними стелажми на підконтейнери.

**Постановка задачі.** Нехай  $\Omega$  – контейнер висотою  $H$ , який має форму паралелепіпеда, циліндра, параболоїда обертання або зрізаного конуса. Контейнер  $\Omega$  заданий у власній нерухомій системі координат  $Oxyz$ , де  $Oz$  – поздовжня вісь симетрії. Вважаємо, що  $\Omega$  розділений горизонтальними стелажми  $S_j$  на підконтейнери  $\Omega^j$ ,  $j \in J_m = \{1, \dots, m\}$ . Відстані між

стелажми  $S_j$  та  $S_{j+1}$  позначимо  $t_j$ ,  $j \in J_m$ ,  $\sum_{j=1}^m t_j = H$ . Початок власної

системи координат  $Oxyz$  розташований у центрі нижньої основи контейнера.

Маємо  $A = \{\mathbb{T}_i, i = 1, \dots, n\}$  – множину однорідних 3D-об'єктів із заданими метричними характеристиками. Кожен об'єкт має висоту  $h_i$  і масу  $m_i$ ,  $i \in J_n = \{1, \dots, n\}$ .

Об'єкт  $\mathbb{T}_i$  задано у власній системі координат  $O_i x_i y_i z_i$ . Розташування об'єкта  $\mathbb{T}_i$  всередині контейнера  $\Omega$  визначається вектором  $u_i = (v_i, z_i, \theta_i)$ ,  $(v_i, z_i)$  – вектор трансляції об'єкта в нерухомій системі координат  $Oxyz$ ,  $\theta_i$  – кут обертання об'єкта  $\mathbb{T}_i$  в площині  $O_i x_i y_i$ , де  $v_i = (x_i, y_i)$ , при цьому значення  $z_i$ ,  $i \in J_n$ , однозначно визначається підконтейнером  $\Omega^j$ ,  $j \in J_m$ , в якому розміщується об'єкт  $\mathbb{T}_i$ :

$$z_i = \sum_{l=1}^j t_{l-1} + h_i, \quad (3.1)$$

де  $j \in J_m$ . Вважаємо, що  $t_0 = 0$  і  $\forall i \in J_n$  існує  $j^* \in J_m$ :  $h_i \leq t_{j^*}$ .

На відміну від задач BLP, розглянутих у працях [123, 124], де апріорі задана вимога на розміщення об'єктів у конкретних підконтейнерах  $\Omega^j$ ,  $j \in J_m$ , у цьому дослідженні формулюється задача балансного компонування об'єктів, яка передбачає генерацію і вибір розбиття множини  $A$  на непорожні підмножини  $A^j$ ,  $j \in J_m$ . Тут  $A^j$  – підмножина об'єктів, які мають бути розміщені на стелажі  $S_j$  всередині підконтейнера  $\Omega^j$ .

Нехай  $J_n^j \subseteq J_n$  – множина індексів об'єктів, що розміщуються в підконтейнері  $\Omega^j$ ,  $j \in J_m$ ,  $\bigcup_{j=1}^m J_n^j = J_n$ ,  $J_n^i \cap J_n^j = \emptyset$ ,  $i \neq j \in J_m$ ;  $k_j = |A^j|$  – кількість об'єктів, що розміщуються в підконтейнері  $\Omega^j$ ,  $k_j > 0$ ,  $j \in J_m$ , при цьому

$$\sum_{j=1}^m k_j = n. \quad (3.2)$$

На сумарну масу об'єктів, що розміщуються в підконтейнері, накладається обмеження:

$$\sum_{i=1}^{k_1} m_i \geq \sum_{i=k_1+1}^{k_1+k_2} m_i \geq \dots \geq \sum_{i=k_1+k_2+\dots+k_{m-1}+1}^n m_i, \quad (3.3)$$

де  $k_j = |A^j| \geq 1, j \in J_m$ .

Крім того, повинні виконуватися такі обмеження розміщення:

$$\text{int } \mathbb{T}_{i_1} \cap \text{int } \mathbb{T}_{i_2} = \emptyset, i_1 < i_2 \in J_n^j, j \in J_m, \quad (3.4)$$

$$\mathbb{T}_i \subset \Omega^j, i \in J_n^j, j \in J_m, \quad (3.5)$$

$$h^j \leq t_j, h^j = \max\{h_i^j, i \in J_n^j\}, j \in J_m. \quad (3.6)$$

Позначимо  $\Omega_A$  систему, утворену в результаті розміщення об'єктів  $\mathbb{T}_i$  сімейства  $A$  в контейнері  $\Omega$ , а як  $O_sXYZ$  – систему координат  $\Omega_A$ , де  $O_s = (x_s(v), y_s(v), z_s(v))$  – центр мас  $\Omega_A$ , а осі  $O_sX \parallel Ox, O_sY \parallel Oy, O_sZ \parallel Oz$ ,

$$x_s(v) = \frac{\sum_{i=1}^n m_i x_i}{M}, y_s(v) = \frac{\sum_{i=1}^n m_i y_i}{M}, z_s = \frac{\sum_{i=1}^n m_i z_i}{M}, \quad (3.7)$$

$M = \sum_{i=1}^n m_i$  – маса системи  $\Omega_A$ .

Як функцію цілі розглядаємо відхилення центра мас  $O_s$  від заданої точки  $(x_0, y_0, z_0)$ .

*Комбінаторна задача балансного компоновання (Combinatorial Balance Layout Problem, CBLP)*. Визначити такий варіант розміщення підмножин  $A^j$ ,

$j \in J_m$  з урахуванням обмежень (3.2)–(3.6) і параметри розміщення  $(x_i^*, y_i^*, z_i^*)$  об'єктів  $T_i$ ,  $i \in J_n$ , за якого функція цілі досягатиме свого мінімального значення.

Вважаємо, що задача має хоча б один припустимий розв'язок.

*Зауваження.* На розміщення об'єктів можуть бути задані також умови балансу (обмеження на осьові і відцентрові моменти системи) і обмеження на допустимі відстані між об'єктами.

**Побудова комбінаторної множини.** Варіанти розбиття множини  $A$  на непорожні підмножини  $A^j$ ,  $j \in J_m$ , визначаються кількістю елементів у кожній підмножині і порядком слідування підмножин. Розглянемо підконтейнери  $\Omega^j$  і відповідні їм множини об'єктів  $A^j$ ,  $j \in J_m$ . Тоді кортеж

натуральних чисел  $(k_1, k_2, \dots, k_m)$  такий, що  $\sum_{j=1}^m k_j = n$ , визначає можливу

кількість  $k_j$  об'єктів у кожному підконтейнері  $\Omega^j$ . Кількість усіх таких кортежів дорівнює кількості композицій числа  $n$  довжиною  $m$  [110], яке становить  $C_{n-1}^{m-1}$ .

Розглянемо тепер, скількома способами можна розкласти  $n$  різних об'єктів з множини  $A$  в  $m$  підконтейнерів  $\Omega^j$ ,  $j \in J_m$ , за умови, що в підконтейнерах знаходиться відповідно  $k_1, k_2, \dots, k_m$  об'єктів, причому множини об'єктів  $A^j$ ,  $j \in J_m$ , усередині відповідних підконтейнерів  $\Omega^j$ ,  $j \in J_m$ , не впорядковані. Не втрачаючи спільності, будемо розрізняти об'єкти з однаковими значеннями метричних характеристик, висоти  $h_i$  і маси  $m_i$  (наприклад, вважати їх такими, що різняться за номером).

Впорядкуємо елементи множини  $A$ . Кожному об'єкту поставимо у відповідність номер підконтейнера, в який він буде поміщений. Отримаємо

кортеж, що складається з  $n$  елементів, які утворюють перестановку з повтореннями з  $m$  номерів  $1, 2, \dots, m$ , в якій перший елемент (номер першого підконтейнера) повторюється  $k_1$  разів, другий –  $k_2$  разів, ..., останній –  $k_m$  разів. Кожна така перестановка визначає свій спосіб розкладання, а їх загальна кількість дорівнює

$$P(n, k_1, k_2, \dots, k_m) = \frac{n!}{k_1! \cdot k_2! \cdot \dots \cdot k_m!}. \quad (3.8)$$

Тоді кількість варіантів розподілу  $n$  різних об'єктів з множини  $A$  по  $m$  підконтейнерам  $\Omega^j$  за умови, що кожен підконтейнер містить хоча б один об'єкт і порядок розміщення об'єктів усередині підконтейнерів не важливий, дорівнює

$$\sum_{k_1+k_2+\dots+k_m=n} P(n, k_1, k_2, \dots, k_m) = \sum_{k_1+k_2+\dots+k_m=n} \frac{n!}{k_1! \cdot k_2! \cdot \dots \cdot k_m!} \quad (3.9)$$

Зауважимо, що кількість доданків у сумі дорівнює кількості композицій числа  $n$  довжиною  $m$ , яке становить  $N = C_{n-1}^{m-1}$ .

Для генерації різних варіантів підмножин  $A^j$ ,  $j \in J_m$ , побудуємо комбінаторну конфігурацію так.

Позначимо  $Pt(n, m)$  множину композицій числа  $n$  довжиною  $m$  (відповідає розподілу  $n$  різних об'єктів з множини  $A$  по  $m$  підконтейнерам  $\Omega^j$ ,  $j \in J_m$ , за умови, що кожен підконтейнер містить хоча б один об'єкт і порядок об'єктів усередині підконтейнерів не важливий). При цьому  $|Pt(n, m)| = N = \left| C_{n-1}^{m-1} \right|$ .

Нехай  $(k_1, k_2, \dots, k_m) \in Pt(n, m)$ ,  $\sum_{j=1}^m k_j = n$ ,  $k_i \geq 1$ ,  $i \in J_m$ .

Уведемо комбінаторну множину  $\mathbb{Q}$ , яка являє собою композиційний образ комбінаторних множин ( $k$ -множину)  $Pt(n, m)$ ;  $C_n^{k_1}$ ,  $C_{n_1}^{k_2}$ ,  $C_{n_2}^{k_3}$ , ...,  $C_{n_{m-1}}^{k_m}$ , породжений множинами  $I_n, I_{n_1}, I_{n_2}, \dots, I_{n_{m-1}}$  [125], де  $n_i = n - k_1 - \dots - k_i$ ,  $i \in J_{m-1}$ ,

$$I_{n_0} = J_n,$$

$$I_{n_1} = I_{n_0} \setminus \{j_1^{n_0}, j_2^{n_0}, \dots, j_{k_1}^{n_0}\}, (j_1^{n_0}, j_2^{n_0}, \dots, j_{k_1}^{n_0}) \in C_n^{k_1},$$

$$I_{n_2} = I_{n_1} \setminus \{j_1^{n_1}, j_2^{n_1}, \dots, j_{k_2}^{n_1}\}, (j_1^{n_1}, j_2^{n_1}, \dots, j_{k_2}^{n_1}) \in C_{n_1}^{k_2},$$

...

$$I_{n_{m-1}} = I_{n_{m-2}} \setminus \{j_1^{n_{m-2}}, j_2^{n_{m-2}}, \dots, j_{k_{m-1}}^{n_{m-2}}\}, (j_1^{n_{m-2}}, j_2^{n_{m-2}}, \dots, j_{k_{m-1}}^{n_{m-2}}) \in C_{n_{m-2}}^{k_{m-1}},$$

$$I_{n_{m-1}} = \{j_1^{n_{m-1}}, j_2^{n_{m-1}}, \dots, j_{k_m}^{n_{m-1}}\}, (j_1^{n_{m-1}}, j_2^{n_{m-1}}, \dots, j_{k_m}^{n_{m-1}}) \in C_{n_{m-1}}^{k_m}.$$

Зазначимо, що

$$I_{n_0} \cup I_{n_1} \cup \dots \cup I_{n_{m-1}} = J_n = \{1, 2, \dots, n\}.$$

$$I_{n_s} \cap I_{n_t} = \emptyset, s \neq t \in J_{m-1}^0 = \{0, 1, \dots, m-1\}.$$

Елемент  $q = (q_1, \dots, q_n) \in \mathbb{Q}$  можна описати так:

$$q = (q_1, \dots, q_{k_1}, q_{k_1+1}, \dots, q_{k_1+k_2}, \dots, q_{k_1+\dots+k_{m-1}}, \dots, q_{k_{m-1}+k_m}),$$

де  $(q_1, \dots, q_{k_1}) = (j_1^{n_0}, j_2^{n_0}, \dots, j_{k_1}^{n_0}) \in C_n^{k_1}$ ,

$$(q_{k_1+1}, \dots, q_{k_1+k_2}) = (j_1^{n_1}, j_2^{n_1}, \dots, j_{k_2}^{n_1}) \in C_{n_1}^{k_2},$$

...

$$\begin{aligned} & (q_{k_1+\dots+k_{m-1}}, \dots, q_{k_{m-1}+k_m}) = \\ & = (j_1^{n_{m-1}}, j_2^{n_{m-1}}, \dots, j_{k_m}^{n_{m-1}}) \in C_{n_{m-1}}^{k_m}. \end{aligned}$$

Потужність множини  $\mathbb{Q}$  визначаємо за формулою (3.9).

Надалі будемо називати елемент  $q = (q_1, \dots, q_n)$  множини  $\mathbb{Q}$  *кортежем розбиття* множини об'єктів  $A$  на підмножини  $A^j$ ,  $j \in J_m$ .

Визначимо вектор змінних задачі CBLP:  $u = (v, z, \theta)$ , де  $v = (v_1, \dots, v_n) \in \mathbf{R}^{2n}$ ,  $\theta = (\theta_1, \dots, \theta_n) \in \mathbf{R}^n$ ,  $v_i = (x_i, y_i) \in \mathbf{R}^2$ ,  $x_i, y_i, \theta_i$  – неперервні змінні,  $z = (z_1, \dots, z_n) \in \mathbf{R}^n$ ,  $z_i$  – дискретні змінні.

Значення змінних  $z_i$ ,  $i = 1, 2, \dots, n$ , визначають у порядку, що заданий елементами  $q = (q_1, \dots, q_n)$  комбінаторної множини  $\mathbb{Q}$ :

$$z_{q_i} = \sum_{l=1}^s t_{l-1} + h_{q_i}, \quad (3.10)$$

де

$$s = \begin{cases} 1, & \text{якщо } i \leq k_1, \\ 2, & \text{якщо } k_1 < i \leq k_1 + k_2, \\ \dots & \\ m, & \text{якщо } k_1 + k_2 + \dots + k_{m-1} < i \leq k_1 + k_2 + \dots + k_m, \end{cases}$$

$$i = 1, 2, \dots, n, q_i \in \{1, 2, \dots, n\}, q = (q_1, \dots, q_n) \in \mathbb{Q}.$$

Отже, множина кортежей розбиття  $\mathbb{Q}$  описує структуру та властивості комбінаторної задачі балансного компонування CBLP.

### 3.2. Математична модель комбінаторної задачі балансного компонування

Побудуємо математичну модель комбінаторної задачі балансного компонування CBLP. Уведемо обмеження на змінні.

*Обмеження розміщення* об'єктів множини  $A$  в контейнері  $\Omega$  виду (3.4), (3.5) описує система нерівностей  $\Upsilon_1(u) \geq 0$ ,  $\Upsilon_2(u) \geq 0$ . Тут  $\Upsilon_1(u) \geq 0$  –

обмеження, яке описує неперетин 3D-об'єктів,  $\Upsilon_2(u) \geq 0$  – обмеження, яке описує включення 3D-об'єктів у контейнер  $\Omega$ , де

$$\Upsilon_1(u) = \min\{\Upsilon_1^j(u), j \in J_m\}, \quad \Upsilon_1^j(u) = \min\{\Upsilon_{q_1 q_2}^j(u_{q_1}, u_{q_2}), q_1 < q_2 \in J_n^j\}, \quad (3.11)$$

$$\Upsilon_2^*(u) = \min\{\Upsilon_2^{*j}(u), j \in J_m\}, \quad \Upsilon_2^{*j}(u) = \min\{\Upsilon_{q_i}^*(u_{q_i}), q_i \in J_n^j\}, \quad (3.12)$$

$\Upsilon_{q_1 q_2}^j(u_{q_1}, u_{q_2})$  – функція, що описує умову неперетину об'єктів  $\mathbb{T}_{q_1}$  і  $\mathbb{T}_{q_2}$ ,

$u_{q_1} = (x_{q_1}, y_{q_1}, z_{q_1}, \theta_{q_1})$ ,  $u_{q_2} = (x_{q_2}, y_{q_2}, z_{q_2}, \theta_{q_2})$ ,  $\Upsilon_{q_i}^*(u_{q_i})$  – функція,

що описує умову неперетину об'єктів  $\mathbb{T}_{q_i}$  і  $\Omega^{*j} = \mathbf{R}^3 / \text{int } \Omega^j$ .

Нехай  $u_1 = (v_1, z_1, \theta_1) \in \mathbf{R}^3$ ,  $u_2 = (v_2, z_2, \theta_2) \in \mathbf{R}^3$ ,  $v_1 = (x_1, y_1)$ ,  $v_2 = (x_2, y_2)$ ,  $x_1, y_1, \theta_1, x_2, y_2, \theta_2$  – неперервні змінні,  $z_1, z_2$  – дискретні змінні,  $u_{12}$  – вектор додаткових змінних.

**Визначення 3.1.** Функція  $\Upsilon_{12}(u_1, u_2)$  називається *D*-phi-функцією для 3D-об'єктів  $\mathbb{T}_1$  і  $\mathbb{T}_2$ , якщо у разі фіксованих значень  $z_1 = z_1^0$  і  $z_2 = z_2^0$  функція  $\Upsilon_{12}(v_1, z_1^0, \theta_1, v_2, z_2^0, \theta_2) \in \text{phi-функцією } \Phi_{12}(v_1, z_1^0, \theta_1, v_2, z_2^0, \theta_2)$  для об'єктів  $\mathbb{T}_1$  і  $\mathbb{T}_2$ .

**Визначення 3.2.** Функція  $\Upsilon'_{12}(u_1, u_2, u_{12})$  називається *D*-квазі-phi-функцією для 3D-об'єктів  $\mathbb{T}_1$  і  $\mathbb{T}_2$ , якщо у разі фіксованих значень  $z_1 = z_1^0$  і  $z_2 = z_2^0$  функція  $\Upsilon'_{12}(v_1, z_1^0, \theta_1, v_2, z_2^0, \theta_2, u'_{12}) \in \text{квазі-phi-функцією } \Phi'_{12}(v_1, z_1^0, \theta_1, v_2, z_2^0, \theta_2, u'_{12})$  для об'єктів  $\mathbb{T}_1$  і  $\mathbb{T}_2$ .

Отже, у співвідношеннях (3.11), (3.12) у разі фіксованих значень  $z_{q_1}$  і  $z_{q_2}$  маємо

$\Upsilon_{q_1 q_2}^j(u_{q_1}, u_{q_2}) \equiv \Phi_{q_1 q_2}^{TT}(u_{q_1}, u_{q_2})$  – phi-функція [87] для об'єктів  $\mathbb{T}_{q_1}$

і  $\mathbb{T}_{q_2}$  або

$\Upsilon_{q_1 q_2}^j(u_{q_1}, u_{q_2}) \equiv \Phi'_{q_1 q_2}{}^{TT}(u_{q_1}, u_{q_2}, u_{q_1 q_2})$  – квазі-pi-функція [25] для

об'єктів  $\mathbb{T}_{q_1}$  і  $\mathbb{T}_{q_2}$ ;

$\Upsilon_{q_i}^*(u_{q_i}) \equiv \Phi_{q_i}^{T\Omega^*j}(u_{q_i})$  – phi-функція для об'єктів  $\mathbb{T}_{q_i}$  і  $\Omega^{*j}$ .

Якщо задані мінімально допустимі відстані між об'єктами, використовуються псевдонормалізовані phi-функції (квазі-pi-функції) для відповідних пар об'єктів [25, 87].

Математичну модель задачі CBLP можна визначити так:

$$F(u^*) = \min F(u) \text{ s.t. } (u, \tau) \in W, \quad (3.13)$$

$$W = \{(u, \tau) \in \mathbf{R}^\sigma : \Upsilon_1(u, \tau) \geq 0, \Upsilon_2^*(u) \geq 0, \mu(u) \geq 0\}, \quad (3.14)$$

де

$$F(u) = d = (x_s(v, z))^2 + (y_s(v, z))^2 + (z_s - z_0)^2,$$

$$u = (v, z, \theta), \quad v = (v_1, \dots, v_n), \quad \theta = (\theta_1, \dots, \theta_n), \quad v_i = (x_i, y_i), \quad i = 1, \dots, n,$$

$z = (z_1, \dots, z_n)$ , функцію  $\Upsilon_1(u, \tau)$  описує співвідношення (3.11) при  $\Xi = \bigcup_{j=1}^m \Xi^j$ ,

$\Xi^j = \{(q_1, q_2) : q_1 < q_2 \in J_n^j\}$ ,  $\tau = (\tau_1, \dots, \tau_s)$  – вектор допоміжних змінних,

$s = |\Xi|$ , функцію  $\Upsilon_2^*(u)$  визначають за формулою (3.12), елементи вектора  $z$  задають співвідношенням (3.10),  $\mu(u) \geq 0$  – умови балансу.

Наприклад, математична модель (3.13), (3.14) для задачі CBLP розміщення циліндрів у циліндричному контейнері набуває вигляду

$$\min d, \text{ s.t. } u = (v, z) \in W,$$

де  $v = (x_1, y_1, \dots, x_n, y_n), z = (z_1, \dots, z_n)$ ,

$$d = \left[ \sum_{i=1}^n m'_i x_i \right]^2 + \left[ \sum_{i=1}^n m'_i y_i \right]^2 + \left[ \sum_{i=1}^n m'_i z_i \right]^2,$$

а множину  $W$  описує система нерівностей:

$$\begin{cases} (x_{q_2} - x_{q_1})^2 + (y_{q_2} - y_{q_1})^2 - (r_{q_2} + r_{q_1})^2 \geq 0, \\ q_1, q_2 \in \Xi^j, j \in J_m, \\ -x_{q_i}^2 - y_{q_i}^2 + (R_{q_i}^z - r_{q_i})^2 \geq 0, \\ q_i \in \Xi^j, j \in J_m. \end{cases}$$

Зауважимо, що  $m'_i = \frac{m_i}{M} = \text{const}$ ,  $M = \sum_{i=1}^n m_i = \text{const}$ .

Математичну модель задачі CBLP можна навести у вигляді задачі змішаного цілочислового програмування (MIP) з використанням булевих змінних. Однак такий підхід, на відміну від (3.13), (3.14), призводить до істотного збільшення кількості дискретних змінних моделі, отже, до зростання розмірності задачі CBLP.

### 3.3. Метод розв'язання комбінаторної задачі балансного компонування

Для розв'язування задач CBLP використовується така стратегія:

1. Випадково генерується множина  $\{q\}$  кортежів розбиття  $q = (q_1, \dots, q_n) \in \mathbb{Q}$  [126].
2. Здійснюється перевірка виконання умов (3.3), (3.6) для кожного з кортежів  $q \in \{q\}$ . Будується підмножина  $\{q'\} \subseteq \{q\}$ , для елементів якої виконуються зазначені умови. Якщо  $\{q'\} = \emptyset$ , то повертаємося до кроку 1.

3. Будується множина допустимих стартових точок  $\{u'_0\}$  для кожного кортежу з множини  $\{q'\}$ .

4. Реалізується пошук локального екстремуму задачі (3.13), (3.14) для кожної стартової точки  $u'_0 \in W$  за фіксованого кортежу  $q'$ .

5. Кращий з отриманих локальних екстремумів для всіх кортежів множини  $\{q'\}$  і допустимих стартових точок множини  $\{u'_0\}$  вибирають як локально-оптимальний розв'язок задачі CBLP.

Наведена стратегія використовує розумний вибір допустимих стартових точок і *NLP-solver* для пошуку локально-оптимальних розв'язків *NP*-складної задачі умовної оптимізації (3.13), (3.14). Для побудови стартових точок з множини допустимих розв'язків запропоновано швидкий алгоритм.

Для зниження обчислювальних витрат (часу і пам'яті) використовується модифікація LOFRT алгоритму, запропонованого в [25]. Такий алгоритм дає змогу звести задачу (3.13), (3.14) великої розмірності з великою кількістю нерівностей до послідовності підзадач зі значно меншою кількістю змінних і нерівностей.

Для розв'язання задач нелінійного програмування використовується IPOPT, доступний на відкритому некомерційному ресурсі (<https://projects.coin-or.org/Ipoprt>). Робота IPOPT базується на методі внутрішньої точки, описаному в [127].

*Алгоритм пошуку допустимих стартових точок.* Алгоритм містить такі кроки для заданого кортежу  $q'$ .

Крок 1. Генеруємо випадково множину точок  $v_i^0 = (x_i^0, y_i^0)$ ,  $i \in I_n$ , що належать відповідним перерізам контейнера. Формуємо вектор  $v^0 = (x_1^0, y_1^0, \dots, x_n^0, y_n^0)$ . Фіксуємо кути повороту  $\theta_i = \theta_i^0 = 0$ ,  $i \in I_n$ .

Крок 2. Нехай  $\lambda = \lambda_i$  – коефіцієнт гомотетії для об'єктів  $A_i$ ,  $i \in I_n$ . Використовуючи очевидні геометричні побудови, визначаємо вектор додаткових змінних  $u'^0$  розмірності  $\tau$  такий, що кожна  $\phi$ -функція або квазі-

phi-функція в (3.14) досягає свого максимального значення за додатковими змінними  $u'^0$  в точці  $(u_\lambda^0, u'^0)$ , де  $u_\lambda^0 = (v^0, \theta^0, \lambda^0)$ ,  $\lambda^0 = 0$ ,  $v^0 = (v_1^0, \dots, v_n^0)$ ,  $\theta^0 = (\theta_1^0, \dots, \theta_n^0)$ .

Крок 3. Обчислюємо  $\alpha^0 = \min\{\Upsilon_1(u_\lambda^0, u'^0), \Upsilon_2(u_\lambda^0)\}$ . Якщо  $\alpha^0 < 0$ , то переходимо до кроку 4, інакше формуємо точку  $u_\alpha^* = (u_\lambda^0, u'^0, \alpha^0)$  і переходимо до кроку 5.

Крок 4. Задаємо  $\lambda = 0$ ,  $\theta_i = \theta_i^0 = 0$ ,  $i \in I_n$ , і використовуємо  $u_\alpha^0 = (u_\lambda^0, u'^0, \alpha^0)$  як стартову точку для розв'язання такої допоміжної задачі нелінійного програмування:

$$\alpha^* = \max \alpha, \text{ s.t. } u_\alpha \in W_\alpha, \quad (3.15)$$

$$W_\alpha = \{u_\alpha \in \mathbb{R}^{3n+\tau+1} : \Upsilon_1(u_\lambda, u') - \alpha \geq 0, \Upsilon_2(u_\lambda) - \alpha \geq 0, -\alpha \geq 0\}, \quad (3.16)$$

де  $u_\alpha = (u_\lambda, u', \alpha)$ .

Якщо  $\alpha^* = 0$ , тоді точку  $u_\alpha^* = (u_\lambda^*, u'^*, \alpha^*)$  глобального максимуму задачі (3.15), (3.16) знайдено і переходимо до кроку 5. Якщо  $\alpha^* < 0$ , то допустиму стартову точку для завдання (3.15), (3.16) не можна знайти, оскільки не виконуються обмеження розміщення при  $\lambda = 0$ . У цьому випадку повертаємося до кроку 1.

Крок 5. Уважаємо, що параметри  $\theta_i$ ,  $i \in I_n$ , – змінні. Генеруємо випадково стартові значення кутів повороту  $\theta_i^* \in [0, 2\pi)$ ,  $i \in I_n$ .

Крок 6. Формуємо допустиму стартову точку  $(u_\lambda^*, u'^*)$ , використовуючи  $u_\alpha^*$ , і розв'язуємо таку допоміжну задачу нелінійного програмування:

$$\lambda^* = \max \lambda, \text{ s.t. } (u_\lambda, u') \in W_\lambda, \quad (3.17)$$

$$W_\lambda = \{(u_\lambda, u') \in \mathbb{R}^{3n+\tau+1} : \Upsilon_1(u_\lambda, u') \geq 0, \Upsilon_2(u_\lambda) \geq 0, 1 - \lambda \geq 0, \lambda \geq 0\}. \quad (3.18)$$

Якщо  $\lambda^* = 1$ , тоді точку  $(u_\lambda^*, u'^*) = (v^*, \theta^*, \lambda^*, u'^*)$  глобального максимуму (3.17), (3.18) знайдено і переходимо до кроку 7. Якщо  $\lambda^* < 1$ , то повертаємося до кроку 1.

Крок 7. Обчислюємо значення  $\mu(v^*, \theta^*)$ . Якщо  $\mu(v^*, \theta^*) < 0$ , то переходимо до кроку 8, інакше – до кроку 9.

Крок 8. Стартуючи з точки  $u_\beta^0 = (v^*, \theta^*, u'^*, \beta^0 = \mu(u^*))$ , розв'язуємо допоміжну задачу:

$$\beta^* = \max \beta, \quad \text{s.t. } u_\beta \in W_\beta, \quad (3.19)$$

$$W_\beta = \{u_\beta \in \mathbb{R}^{3n+1} : \Upsilon_1(u, u') \geq 0, \Upsilon_2(u) \geq 0, \mu(u) - \beta \geq 0, -\beta \geq 0\}, \quad (3.20)$$

де  $\beta$  – нев'язка,  $u_\beta = (u, u', \beta)$ ,  $u = (v, \theta)$ .

Якщо  $\beta^* = 0$ , тоді точку  $u_\beta^* = (v^*, \theta^*, u'^*, \beta^*)$  глобального максимуму задачі (3.19), (3.20) знайдено, переходимо до кроку 9. Якщо  $\beta^* < 0$ , то повертаємося до кроку 1.

Крок 9. Формуємо допустиму стартову точку  $u^0 = (v^*, \theta^*, u'^*) \in W$  для задачі CBLP.

### 3.4. Результати числових експериментів

Приклад 3.1. Розглядаємо задачу балансного компонування циліндрів  $\mathcal{C}_i, i=1, \dots, 8$ , у циліндричному контейнері, розділеному двома круговими стелажми на підконтейнери з метою мінімізації відхилення центра мас системи  $\Omega_A$  від точки  $(x_0, y_0, z_0)$ .

Нехай  $n = 8$ ,  $m = 3$ ,  $H = 6$ ,  $R = 2,5$ ,  $t_1 = t_2 = 2$ ,  $(x_0, y_0, z_0) = (0, 0, 3)$ .

Маси і метричні характеристики (радіуси і піввисоти) циліндрів  $C_i, i = 1, \dots, 8$ , задано в таб. 3.1

Результати числових експериментів для прикладу 3.1.

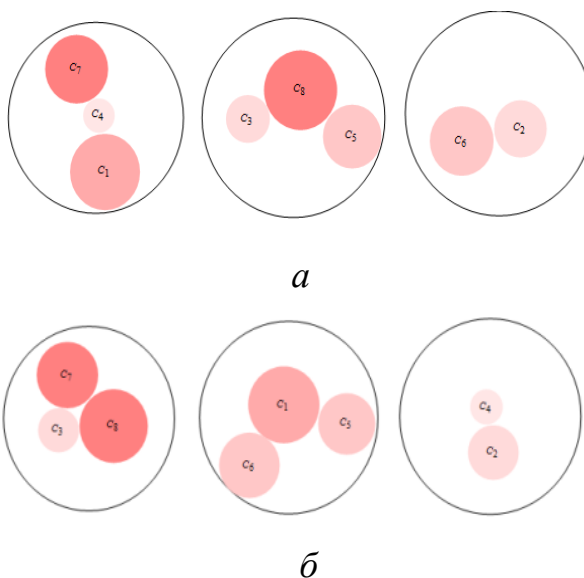
Значення функції цілі  $d^* = 0,3851$  при  $q^1 = (1, 4, 7, | 3, 5, 8, | 2, 6)$ .

Значення функції цілі  $d^* = 0,8847$  при  $q^2 = (3, 7, 8, | 1, 5, 6, | 2, 4)$ .

Таблиця 3.1. Вихідні дані для циліндрів

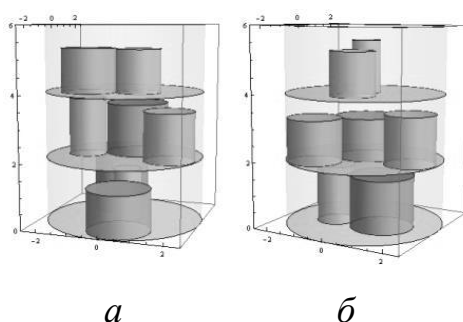
	$C_1$	$C_2$	$C_3$	$C_4$	$C_5$	$C_6$	$C_7$	$C_8$
$m_i$	4	2	2	1	3	3	5	5
$r_i$	1	0,7	0,6	0,45	0,8	0,85	0,9	1
$h_i$	1,27	1,3	1,77	1,57	1,49	1,32	1,96	1,59

На рис. 3.1 наведено проєкції локально-оптимальних розміщень циліндрів у підконтейнерах, що відповідають кортежам  $q^1$  і  $q^2$ .



**Рис. 3.1.** Проєкції локально-оптимальних розміщень циліндрів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають:  $a$  – кортежу  $q^1$ ;  $b$  – кортежу  $q^2$

На рис. 3.2 наведено локально-оптимальні розміщення циліндрів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають кортежам  $q^1$  і  $q^2$ .

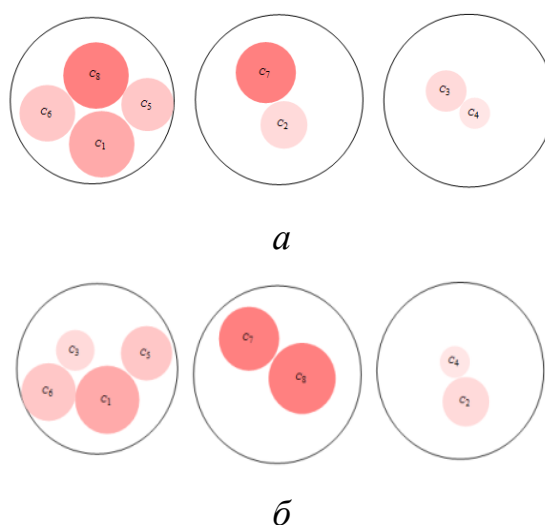


**Рис. 3.2.** Локально-оптимальні розміщення циліндрів у підконтейнерах, що відповідають:  $a$  – кортежу  $q^1$ ;  $b$  – кортежу  $q^2$

На рис. 3.3 наведено проєкції локально-оптимальних розміщень циліндрів у підконтейнерах, що відповідають кортежам  $q^3$  і  $q^4$ .

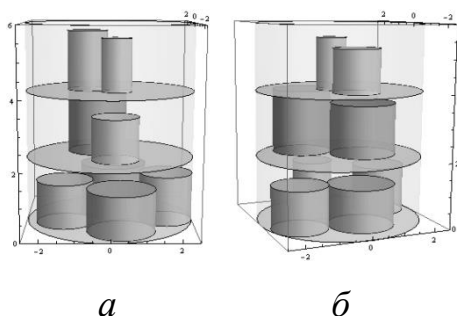
Значення функції цілі  $d^*=1,3938$  при  $q^3 = (1, 5, 6, 8, | 2, 7, | 3, 4)$ .

Значення функції цілі  $d^*=1,8847$  при  $q^4 = (1, 3, 5, 6, | 7, 8, | 2, 4)$ .



**Рис. 3.3.** Проєкції локально-оптимальних розміщень циліндрів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають:  $a$  – кортежу  $q^3$ ;  $b$  – кортежу  $q^4$

На рис. 3.4 наведено локально-оптимальні розміщення циліндрів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають кортежам  $q^3$  і  $q^4$ .



**Рис. 3.4.** Локально-оптимальні розміщення циліндрів в підконтейнерах, що відповідають:  $a$  – кортежу  $q^3$ ;  $b$  – кортежу  $q^4$

Найкращий результат  $d^*=0,3851$  отримано для прикладу 3.1 при  $q^1 = (1, 4, 7, |3, 5, 8, |2, 6)$ .

Приклад 3.2. Розглядаємо задачу балансного компоновання сімейства 3D-об'єктів (куль  $S_i$ , прямих колових циліндрів  $C_i$ , торів  $Q_i$ , сфероциліндрів  $S_{C_i}$ , прямих прямокутних паралелепіпедів  $P_i$  і правильних призм  $K_i$ ) в циліндричному контейнері з урахуванням умови балансу (осьові і відцентрові моменти системи) і обмежень на мінімально допустимі відстані між об'єктами, а також між об'єктами і межею контейнера, з метою мінімізації відхилення центра мас системи  $\Omega_A$  від точки  $(x_0, y_0, z_0)$ .

Нехай  $n = 20$ ,  $m = 3$ ,  $H = 1$ ,  $R = 0.45$ ,  $t_1 = t_2 = 0.35$ ,  $A = \{S_i, i = 1, \dots, 4, C_i, i = 5, \dots, 8, Q_i, i = 9 \dots 12, S_{C_i}, i = 13, \dots, 16, P_i, i = 17, 18, 19, K_{20}\}$ ,  $(x_0, y_0, z_0) = (0, 0, 0.5)$ ,  $\{m_i, i = 1, \dots, 20\} = \{20.944, 15.2681, 27.8764, 34.5575, 63.7115, 41.8146, 30.4106, 28.4245, 49.9649, 24.8714, 38.6888, 26.2637, 20.7764, 17.2159, 16.8756, 52.8, 52.8, 52.8, 23.1489\}$ ; радіуси куль  $S_i, i = 1, \dots, 4$ , радіуси  $r_1 = 0.1$ ,  $r_2 = 0.09$ ,  $r_3 = 0.11$ ,  $r_4 = 0.11$ ; радіуси і піввисоти циліндрів  $C_i, i = 5, \dots, 8$ , радіуси  $r_5 = 0.1$ ,  $h_5 = 0.11$ ,  $r_6 = 0.13$ ,  $h_6 = 0.12$ ,  $r_7 = 0.11$ ,

$h_7 = 0.11$ ,  $r_8 = 0.11$ ,  $h_8 = 0.08$ ; відстані від центрів твірних кіл до осей обертання і піввисот торів  $Q_i, i = 9 \dots 12$ , дорівнюють  $r_9 = 0.08$ ,  $h_9 = 0.07$ ,  $r_{10} = 0.09$ ,  $h_{10} = 0.075$ ,  $r_{11} = 0.07$ ,  $h_{11} = 0.06$ ,  $r_{12} = 0.08$ ,  $h_{12} = 0.07$ ; радіуси, піввисоти циліндрів і висоти кульових сегментів для сфероциліндрів  $S_{C_i}, i = 13, \dots, 16$ , дорівнюють  $r_{13} = 0.1$ ,  $h_{13} = 0.05$ ,  $l_{13} = 0.07$ ,  $r_{14} = 0.05$ ,  $h_{14} = 0.05$ ,  $l_{14} = 0.08$ ,  $r_{15} = 0.08$ ,  $h_{15} = 0.05$ ,  $l_{15} = 0.06$ ,  $r_{16} = 0.08$ ,  $h_{16} = 0.04$ ,  $l_{16} = 0.07$ ; паралелепіпеди задані півширинами, півдовжинами і піввисотами, відповідно,  $w_{17} = 0.11$ ,  $l_{17} = 0.1$ ,  $h_{17} = 0.12$ ,  $w_{18} = 0.11$ ,  $l_{18} = 0.1$ ,  $h_{18} = 0.12$ ,  $w_{19} = 0.11$ ,  $l_{19} = 0.1$ ,  $h_{19} = 0.12$ ; правильна шестикутна призма задана довжиною сторони основи  $r_{20} = 0.09$  і піввисотою  $h_{20} = 0.11$ . Мініміально допустимі відстані між об'єктами, а також між об'єктами і межею контейнера дорівнюють  $\rho_i^- = \rho_{ij}^- = 0.02$ ,  $i < j = 1, \dots, 20$ . Допустимі значення осьових і відцентрових моментів інерції становлять  $\Delta J_X = \Delta J_Y = 170$ ,  $\Delta J_Z = 150$  і  $\Delta J_{XY} = \Delta J_{YZ} = \Delta J_{XZ} = 0$  відповідно.

Результати числових експериментів для прикладу 3.2.

Значення функції цілі  $d^* = 0,0019$  при

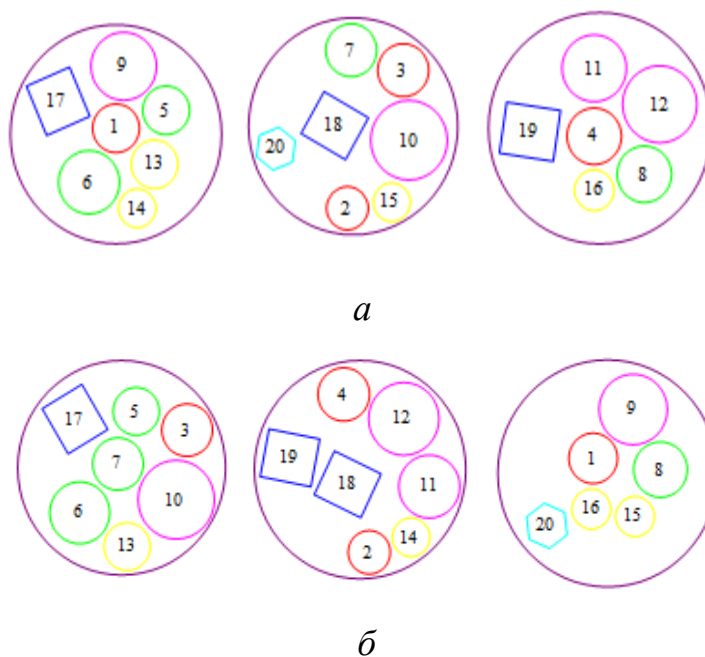
$$q^1 = (1, 5, 6, 9, 13, 14, 17 | 2, 3, 7, 10, 15, 18, 20 | 4, 8, 11, 12, 16, 19).$$

Значення функції цілі  $d^* = 0,0056$  при

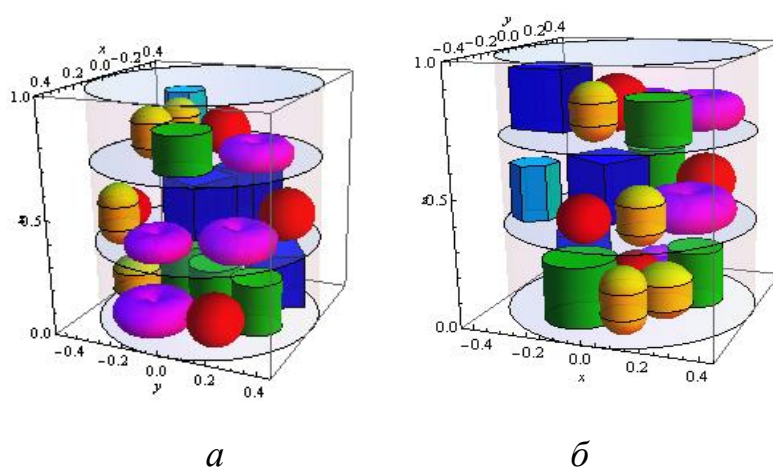
$$q^2 = (3, 5, 6, 7, 10, 13, 17 | 2, 4, 11, 12, 14, 18, 19 | 1, 8, 9, 15, 16, 20).$$

На рис. 3.5 наведено проєкції локально-оптимальних розміщень об'єктів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають кортежам  $q^1$  і  $q^2$ .

На рис. 3.6 наведено локально-оптимальні розміщення об'єктів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають кортежам  $q^1$  і  $q^2$ .



**Рис. 3.5.** Проекції локально-оптимальних розміщень об'єктів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають:  $a$  – кортежу  $q^1$ ;  $b$  – кортежу  $q^2$



**Рис. 3.6.** Локально-оптимальні розміщення об'єктів у підконтейнерах, що відповідають:  $a$  – кортежу  $q^1$ ;  $b$  – кортежу  $q^2$

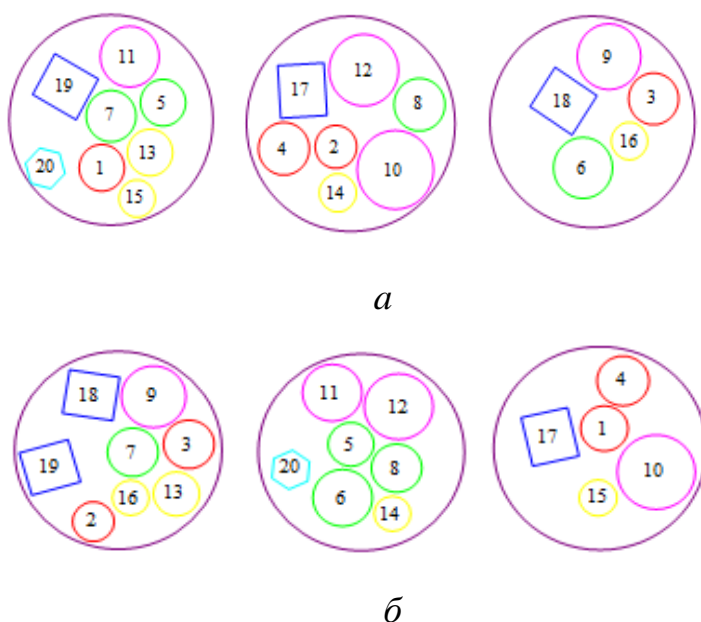
Значення функції цілі  $d^* = 0,0018$  при

$$q^3 = (1, 5, 7, 11, 13, 15, 19, 20 | 2, 4, 8, 10, 12, 14, 17 | 3, 6, 9, 16, 18).$$

Значення функції цілі  $d^* = 0,0030$  при

$$q^4 = (2, 3, 7, 9, 13, 16, 18, 19 | 5, 6, 8, 11, 12, 14, 20 | 1, 4, 10, 15, 17).$$

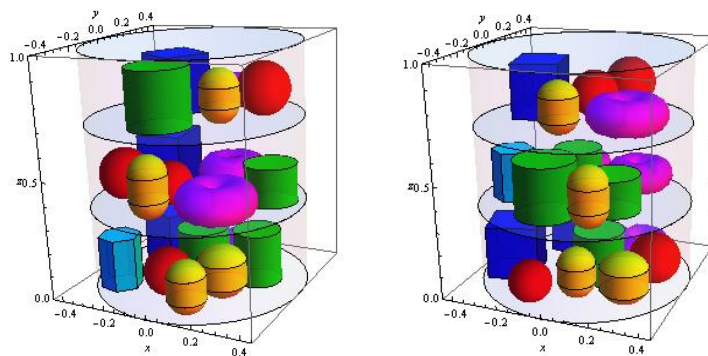
На рис. 3.7 наведено проєкції локально-оптимальних розміщень об'єктів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають кортежам  $q^3$  і  $q^4$ .



**Рис. 3.7.** Проєкції локально-оптимальних розміщень об'єктів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають:  $a$  – кортежу  $q^3$ ;  $b$  – кортежу

$$q^4$$

На рис. 3.8 наведено локально-оптимальні розміщення об'єктів у підконтейнерах  $\Omega^1, \Omega^2, \Omega^3$ , що відповідають кортежам  $q^3$  і  $q^4$ .



**Рис. 3.8.** Локально-оптимальні розміщення об'єктів у підконтейнерах, що відповідають:  $a$  – кортежу  $q^3$ ;  $b$  – кортежу  $q^4$

Найкращий результат  $d^* = 0,0018$  отримано для прикладу 3.2 при

$$q^3 = (1, 5, 7, 11, 13, 15, 19, 20 | 2, 4, 8, 10, 12, 14, 17 | 3, 6, 9, 16, 18).$$

3.5. Комбінаторна задача балансного компоунання зі спеціальними геометричними обмеженнями та умовами балансу

У підрозділі розглянуто пакування набору опуклих геометричних об'єктів у контейнер з опорними полицями (стелажми) з урахуванням спеціальних геометричних та механічних (поведінкових) обмежень. Ця задача є розвитком задач балансного пакування, досліджених у [27, 123, 124], та, зокрема, побудови комбінаторної задачі балансного компоунання, наведеної в п. 3.1–3.4.

Опуклий контейнер може мати форму циліндра, параболоїда або зрізаного кругового конуса. Як об'єкти розміщення розглядають набір опуклих тривимірних об'єктів, включаючи (але не обмежуючись цим) циліндри, кулі, диски (опукла оболонка тора), кубоїди та правильні призми з багатокутною основою. Задача пакування, що розглядають, враховує мінімальні та максимально допустимі відстані, а також комбінаторні характеристики

розміщення об'єктів усередині підконтейнерів (об'єкт може бути розміщений лише на або під опорними полицями).

Метод  $\phi$ -функцій [25, 45, 128–130] використовується для аналітичного опису обмежень розміщення: неперетину об'єктів, утримання об'єктів у контейнері, обмеження відстані, що є спеціальними вимогами до розташування об'єктів у підконтейнерах. Механічні обмеження, що відповідають рівновазі, моментам інерції та стійкості, викладені аналогічно [124]. Також розглянуто спеціальні комбінаторні характеристики (див. п. 3.1), пов'язані з розбиттям об'єктів на підмножини, які розміщуються в підконтейнерах. Побудовано математичну модель задачі пакування, запропоновано та чисельно перевірено підхід до її розв'язання.

**Постановка задачі.** Нехай  $\Omega \subset \mathbb{R}^3$  є опуклим контейнером (наприклад, циліндром, параболоїдом, зрізаним конусом). Контейнер  $\Omega$  поділяється на підконтейнери  $\Omega^k$ ,  $k = 1, 2, \dots, m$ , круговими опорними полицями (стелажми)  $S_k$ ,  $k = 1, 2, \dots, m + 1$ , за умови, що  $S_1$  є основою  $\Omega$ . Відстань  $t_k$  між опорними стелажми  $S_k$  і  $S_{k+1}$  задано.

Нехай  $A = \{\mathbb{T}_i, i \in J_n\}$  – набір опуклих об'єктів. Кожний об'єкт  $\mathbb{T}_i$  висотою  $h_i$  (“вертикальна довжина” об'єкта) та вагою  $m_i$  визначається в його локальній системі координат  $O_i x_i y_i z_i$ ,  $i \in J_n$ . Розташування об'єкта  $\mathbb{T}_i$  всередині контейнера  $\Omega$  визначається вектором  $u_i = (v_i, z_i, \theta_i)$ , де  $(v_i, z_i)$  є вектор трансляції у фіксованій системі координат  $Oxyz$ ,  $\theta_i$  – кут повороту об'єкта  $\mathbb{T}_i$  в площині  $O_i x_i y_i$ , де  $v_i = (x_i, y_i)$ . Значення  $z_i$ ,  $i \in J_n$ , однозначно визначається підконтейнером  $\Omega^j$ ,  $j \in J_m$ , в якому об'єкт  $\mathbb{T}_i$  має бути розміщений. Кожен об'єкт  $\mathbb{T}_i$  може переміщатися по відповідному опорному стелажу.

Можуть бути задані мінімальна  $\rho_{ij}^-$  та максимальна  $\rho_{ij}^+ \geq \rho_{ij}^-$  допустимі відстані між об'єктами  $\mathbb{T}_i, \mathbb{T}_j$ ,  $j > i \in J_n$ . Мініально допустиму відстань  $\rho_i^-$  між об'єктом  $\mathbb{T}_i$ ,  $i \in J_n$ , і боковою поверхнею контейнера  $\Omega$  також можна визначити. Встановлюємо  $\rho_{ij}^- = 0$  (або  $\rho_{ij}^+ = \varpi$ ), якщо мінімальна (та / або максимальна) допустима відстань між об'єктами  $\mathbb{T}_i, \mathbb{T}_j$  не визначені,  $j > i \in J_n$ . Тут  $\varpi$  – задане досить велике число. Також встановлюємо  $\rho_i^- = 0$ , якщо мініально допустиму відстань між об'єктом  $\mathbb{T}_i$  і боковою поверхнею контейнера  $\Omega$  не визначено.

Обмеження розміщення у задачі пакування можна записати так:  $\rho_{ij}^- \leq \text{dist}(\mathbb{T}_i, \mathbb{T}_j) \leq \rho_{ij}^+$ ,  $j > i \in J_n$ , і  $\text{dist}(\mathbb{T}_i, \Omega^*) \geq \rho_i^-$ ,  $i \in J_n$ , де  $\Omega^* = \mathbb{R}^3 \setminus \text{int } \Omega$ .

Позначимо підмножину об'єктів, які мають бути розміщені на опорному стелажі  $S_j$  всередині  $\Omega^j$  як  $A^j$ ,  $j \in J_m$ .

Для розміщення об'єкта  $\mathbb{T}_i$ ,  $i \in J_n$ , у підконтейнер  $\Omega^j$ ,  $j \in J_m$ , мають бути виконані такі умови:

$$z_i = \sum_{l=1}^j t_{l-1} + h_i. \quad (3.21)$$

Вважаємо, що  $t_0 = 0$  та  $\forall i \in J_n$  існує  $j^* \in J_m$ :  $h_i \leq t_{j^*}$ .

Нехай  $J_n^j \subseteq J_n$  – набір індексів об'єктів, які поміщаються в підконтейнер  $\Omega^j$ ,  $j \in J_m$ ,

$$\bigcup_{j=1}^m J_n^j = J_n, \quad J_n^i \cap J_n^j = \emptyset, \quad i \neq j \in J_m; \quad (3.22)$$

$k_j = |A^j|$  – кількість об'єктів, які поміщаються в підконтейнер  $\Omega^j$ ,  
 $k_j > 0, j \in J_m$ ,

$$\sum_{j=1}^m k_j = n, \quad (3.23)$$

$$\rho_{ij}^- \leq \text{dist}(\mathbb{T}_{i_1}, \mathbb{T}_{i_2}) \leq \rho_{ij}^+, i_1 < i_2 \in J_n^j, j \in J_m, \quad (3.24)$$

$$\text{dist}(\mathbb{T}_i, \Omega^{j*}) \geq \rho_i^-, i \in J_n^j, j \in J_m, \quad (3.25)$$

$$h^j \leq t_j, h^j = \max\{h_i^j + \rho_i^*, i \in J_n^j\}, j \in J_m. \quad (3.26)$$

Також потрібно враховувати варіанти розбиття набору об'єктів  $A$  на непорожні підмножини  $A^j, j \in J_m$ , та визначити відповідні параметри розміщення  $u_i = (v_i, z_i, \theta_i)$  об'єктів  $\mathbb{T}_i, i \in J_n$ , із урахуванням співвідношень (3.22) – (3.26).

Позначимо  $u = (v, z, \theta), v = (v_1, \dots, v_n), \theta = (\theta_1, \dots, \theta_n), v_i = (x_i, y_i), i \in J_n, z = (z_1, \dots, z_n)$ .

Позначимо систему, отриману в результаті розміщення об'єктів  $\mathbb{T}_i$  множини  $A$  в контейнері  $\Omega$ , як  $\Omega_A$ . Відповідно, система координат  $\Omega_A$  позначається як  $O_sXYZ$ , де  $O_s = (x_s(v), y_s(v), z_s(v))$  – центр ваги  $\Omega_A$ :

$$x_s(v) = \frac{\sum_{i=1}^n m_i x_i}{M}, y_s(v) = \frac{\sum_{i=1}^n m_i y_i}{M}, z_s = \frac{\sum_{i=1}^n m_i z_i}{M}. \quad (3.27)$$

Тут  $M = \sum_{i=1}^n m_i$  – вага системи  $\Omega_A$  та  $O_sX \parallel Ox, O_sY \parallel Oy, O_sZ \parallel Oz$ .

Найчастіше використовуваними цільовими функціями (див., наприклад, [27, 105, 118–120]) є: 1) розміри контейнера  $\Omega$ ; 2) відхилення центра ваги  $\Omega_A$  від заданої точки; 3) моменти інерції  $\Omega_A$ .

Задача пакування може бути сформульована так:

*Комбінаторна задача балансного компоновання зі спеціальними геометричними обмеженнями та умовами балансу.* Упакувати набір об'єктів  $\mathbb{T}_i$ ,  $i \in J_n$ , всередині контейнера  $\Omega$  так, щоб задана цільова функція досягала екстремального значення з урахуванням обмежень розміщення та умов балансу.

Для аналітичного опису обмежень розміщення використовується метод  $\phi$ -функцій (див. п. 1.5).

Для генерації підмножин  $A^j$ ,  $j \in J_m$ , використовуємо спеціальний алгоритм, описаний у [27]. Цей алгоритм базується на комбінаторних конфігураціях, що називаються  $Y_2(u) \geq 0$ -множинами.

Значення змінних  $z_i$ ,  $i \in J_n$ , визначаються в порядку, заданому елементами  $q(\mathbb{k})$  комбінаторної множини  $Q(\mathbb{k})$ , визначеної в п. 3.1:

$$z_{q_i} = \sum_{l=1}^g t_{l-1} + h_{q_i}, \text{ де}$$

$$g = \begin{cases} 1, \text{ якщо } i \leq k_1, \\ 2, \text{ якщо } k_1 < i \leq k_1 + k_2, \\ \dots \\ m, \text{ якщо } k_1 + k_2 + \dots + k_{m-1} < i \leq k_1 + k_2 + \dots + k_m, \end{cases}$$

$$i = 1, 2, \dots, n, q_i \in \{1, 2, \dots, n\}, q(\mathbb{k}) \in Q(\mathbb{k}).$$

Розглянемо механічні характеристики системи  $\Omega_A$ .

Обмеження рівноваги визначають за такою системою нерівностей:

$$\begin{cases} \mu_{11}(u, p) = \min\{-(x_s(u) - x_e) + \Delta x_e, (x_s(u) - x_e) + \Delta x_e\} \geq 0 \\ \mu_{12}(u, p) = \min\{-(y_s(u) - y_e) + \Delta y_e, (y_s(u) - y_e) + \Delta y_e\} \geq 0, \\ \mu_{13}(u, p) = \min\{-(z_s(u) - z_e) + \Delta z_e, (z_s(u) - z_e) + \Delta z_e\} \geq 0 \end{cases}$$

де  $(x_e, y_e, z_e)$  – очікувані координати  $O_s$ ,  $(\Delta x_e, \Delta y_e, \Delta z_e) \in$  допустимими відхиленнями від точки  $(x_e, y_e, z_e)$ .

Обмеження для моментів інерції визначають так:

$$\begin{cases} \mu_{21}(u, p) = -J_X(u, p) + \Delta J_X \geq 0 \\ \mu_{22}(u, p) = -J_Y(u, p) + \Delta J_Y \geq 0 \\ \mu_{23}(u, p) = -J_Z(u, p) + \Delta J_Z \geq 0 \end{cases}$$

де  $J_X(u, p), J_Y(u, p), J_Z(u, p)$  – моменти інерції системи  $\Omega_A$  відносно осей системи координат  $O_sXYZ$  та  $\Delta J_X, \Delta J_Y, \Delta J_Z$  – допустимі відхилення для значень  $J_X(u, p), J_Y(u, p), J_Z(u, p)$ , де

$$J_X(u, p) = J_{x_0} + \sum_{i=1}^n (J_{x_i} \cos^2 \theta_i + J_{y_i} \sin^2 \theta_i) + \sum_{i=1}^n (y_i^2 + z_i^2) m_i - M(y_s^2 + z_s^2),$$

$$J_Y(u, p) = J_{y_0} + \sum_{i=1}^n (J_{x_i} \sin^2 \theta_i + J_{y_i} \cos^2 \theta_i) + \sum_{i=1}^n (x_i^2 + z_i^2) m_i - M(x_s^2 + z_s^2),$$

$$J_Z(u, p) = \sum_{i=0}^n J_{z_i} + \sum_{i=1}^n (y_i^2 + z_i^2) m_i - M(x_s^2 + y_s^2),$$

$J_{x_0}, J_{y_0}, J_{z_0}$  – моменти інерції  $\Omega$  відносно осей системи координат  $Oxyz$  та  $J_{x_i}, J_{y_i}, J_{z_i}, i \in I_n$  – моменти інерції об'єкта  $A_i$  відносно осей системи координат  $O_i x_i y_i z_i$ .

Обмеження стабільності визначають за такою системою нерівностей:

$$\begin{cases} \mu_{31}(u, p) = \min\{-J_{XY}(u, p) + \Delta J_{XY}, J_{XY}(u, p) + \Delta J_{XY}\} \geq 0 \\ \mu_{32}(u, p) = \min\{-J_{YZ}(u, p) + \Delta J_{YZ}, J_{YZ}(u, p) + \Delta J_{YZ}\} \geq 0 \\ \mu_{33}(u, p) = \min\{-J_{XZ}(u, p) + \Delta J_{XZ}, J_{XZ}(u, p) + \Delta J_{XZ}\} \geq 0 \end{cases}$$

де  $J_{XY}(u, p), J_{YZ}(u, p), J_{XZ}(u, p)$  – моменти інерції  $\Omega_A$  відносно осей системи координат  $O_sXYZ$  та  $\Delta J_{XY}, \Delta J_{YZ}, \Delta J_{XZ}$  – допустимі відхилення для значень  $J_{XY}(u, p), J_{YZ}(u, p), J_{XZ}(u, p)$  відповідно,

$$J_{XY}(u, p) = \frac{1}{2} \sum_{i=1}^n (J_{x_i} - J_{y_i}) \sin 2\theta_i + \sum_{i=1}^n x_i y_i m_i - M x_s y_s,$$

$$J_{YZ}(u, p) = \sum_{i=1}^n y_i z_i m_i - M y_s z_s, \quad J_{XZ}(u) = \sum_{i=1}^n x_i z_i m_i - M x_s z_s.$$

**Математична модель.** Математична модель комбінаторної задачі балансного компоновання зі спеціальними геометричними обмеженнями та умовами балансу можна навести у формі

$$\min F(p, u, \tau), \quad (u, p) \in W, \quad (3.28)$$

$$W = \{(u, p, \tau) \in \mathbb{R}^\xi : \Upsilon_1(u, \tau) \geq 0, \Upsilon_2^*(u, p) \geq 0, \mu(u, p) \geq 0, \zeta(u, p) \geq 0\}, \quad (3.29)$$

де  $F(p, u, \tau) = \alpha_1 F_1(p, u, \tau) + \alpha_2 F_2(p, u, \tau) + \dots + \alpha_k F_k(p, u, \tau)$ ,

$$\sum_{l=1}^k \alpha_l = 1, \alpha_l \geq 0, \quad u = (v, z, \theta), \quad v = (v_1, \dots, v_n), \quad \theta = (\theta_1, \dots, \theta_n), \quad v_i = (x_i, y_i),$$

$i \in J_n, z = (z_1, \dots, z_n), \tau$  є вектором усіх допоміжних змінних для квазі-phi-функцій,

$$\Upsilon_1(u, \tau) = \min\{\Upsilon_1^j(u, \tau), j \in J_m\}, \quad (3.30)$$

$$\Upsilon_1^j(u, \tau) = \min\{\Upsilon_{q_1 q_2}^{-j}(u_{q_1}, u_{q_2}, u_{q_1 q_2}), q_1 < q_2 \in J_n^j, (q_1, q_2) \in \Xi_-^k, \\ \Upsilon_{q_1 q_2}^{+j}(u_{q_1}, u_{q_2}, u_{q_1 q_2}), q_1 < q_2 \in J_n^j, (q_1, q_2) \in \Xi_+^k, k=1, 2, \dots, m\},$$

$$\tau = (u_{q_1 q_2}^-, (q_1, q_2) \in \Xi_-^k, u_{q_1 q_2}^+, (q_1, q_2) \in \Xi_+^k, k=1, \dots, m),$$

$$\Xi_-^k = \{(i, j) : |z_i - z_j| < h_i + h_j + \rho_{ij}^-, i < j \in I^k\},$$

$$\Xi_+^k = \{(i, j) : \rho_{ij}^+ < \varpi, i < j \in I^k\},$$

$$\Upsilon_{q_1 q_2}^{-j} \in \{\widehat{\Phi}_{q_1 q_2}^{-j}, \widehat{\Phi}'_{q_1 q_2}{}^{-j}\}, (q_1, q_2) \in \Xi_-^k,$$

$$\Upsilon_{q_1 q_2}^{+j} \in \{\widehat{\Phi}_{q_1 q_2}^{+j}, \widehat{\Phi}'_{q_1 q_2}{}^{+j}\}, (q_1, q_2) \in \Xi_+^k,$$

$\Upsilon_{q_1 q_2}^j(u_{q_1}, u_{q_2}, u_{q_1 q_2})$  – функція, яка відповідає за обмеження відстані між об'єктами  $\mathbb{T}_{q_1}$  та  $\mathbb{T}_{q_2}$ ,  $u_{q_1} = (x_{q_1}, y_{q_1}, z_{q_1}, \theta_{q_1})$ ,  $u_{q_2} = (x_{q_2}, y_{q_2}, z_{q_2}, \theta_{q_2})$ ,

$$\Upsilon_2^*(u, p) = \min\{\Upsilon_2^{*j}(u, p), j \in J_m\}, \Upsilon_2^{*j}(u, p) = \min\{\Upsilon_{q_i}^*(u_{q_i}, p), q_i \in J_n^j\}, (3.31)$$

$\Upsilon_{q_i}^*(u_{q_i}, p)$  – функція, яка описує обмеження відстані для об'єктів  $\mathbb{T}_{q_i}$  і  $\Omega^{*j} = \mathbb{R}^3 / \text{int } \Omega^j$ .

Отже, у виразах (3.30), (3.31) для фіксованих  $z_{q_1}$  і  $z_{q_2}$  маємо:

$\Upsilon_{q_1 q_2}^j(u_{q_1}, u_{q_2})$  є  $\rho$ -функцією [129]  $\Phi_{q_1 q_2}^{\mathbb{T}\mathbb{T}}(u_{q_1}, u_{q_2})$  для об'єктів  $\mathbb{T}_{q_1}$  та  $\mathbb{T}_{q_2}$  або квазі- $\rho$ -функцією [25, 128]  $\Phi_{q_1 q_2}^{\mathbb{T}\mathbb{T}}(u_{q_1}, u_{q_2}, u_{q_1 q_2})$  для об'єктів  $\mathbb{T}_{q_1}$  та  $\mathbb{T}_{q_2}$ ;  $\Upsilon_{q_i}^*(u_{q_i})$  є  $\rho$ -функцією  $\Phi_{q_i}^{\mathbb{T}\Omega^{*j}}(u_{q_i})$  для об'єктів  $\mathbb{T}_{q_i}$  і  $\Omega^{*j}$ .

У (3.29) функція  $\mu(u, p) = \min\{\mu_s(u, p), s=1, 2, 3\}$  відповідає за поведінкові обмеження, де

$$\mu_1(u, p) = \min\{\mu_{11}(u, p), \mu_{12}(u, p), \mu_{13}(u, p)\},$$

$$\mu_2(u, p) = \min\{\mu_{21}(u, p), \mu_{22}(u, p), \mu_{23}(u, p)\},$$

$$\mu_3(u, p) = \min\{\mu_{31}(u, p), \mu_{32}(u, p), \mu_{33}(u, p)\}.$$

У (3.29)  $\zeta(u, p) \geq 0$  є системою додаткових обмежень на метричні характеристики контейнера  $\Omega$  та параметри розміщення об'єктів.

Залежно від вибору цільових функцій  $F_1(p, u), F_2(p, u), \dots, F_k(p, u)$  можуть бути згенеровані різні варіанти математичної моделі (3.28), (3.29).

Розглянемо кілька варіантів математичної моделі (3.28), (3.29):

$$\text{а) } F(p, u) = d = (x_s(p, u) - x_e)^2 + (y_s(p, u) - y_e)^2 + (z_s(p, u) - z_e)^2,$$

$$W_1 = \{(p, u, \tau) \in \mathbb{R}^\xi : Y_1(u, \tau) \geq 0, Y_2(p, u) \geq 0, \mu_2(p, u) \geq 0, \mu_3(p, u) \geq 0\};$$

$$\text{б) } F(p, u, \tau) = \alpha_1 F_1(p, u, \tau) + \alpha_2 F_2(p, u), F_1(p, u, \tau) = f(p), F_2(p, u) = d,$$

$$W_2 = \{(p, u, \tau) \in \mathbb{R}^\xi : Y_1(u, \tau) \geq 0, Y_2(p, u) \geq 0, \mu_2(p, u) \geq 0, \mu_3(p, u) \geq 0\};$$

$$\text{в) } F(p, u) = \alpha_1 F_1(p, u, \tau) + \alpha_2 F_2(p, u) + \alpha_3 F_3(p, u),$$

$$F_1(p, u) = J_X(p, u), F_2(p, u) = J_Y(p, u), F_3(p, u) = J_Z(p, u),$$

$$W_3 = \{(p, u, \tau) \in \mathbb{R}^\xi : Y_1(u, \tau) \geq 0, Y_2(p, u) \geq 0, \mu_1(p, u) \geq 0, \mu_3(p, u) \geq 0\}.$$

**Стратегія розв'язання.** Задача (3.28), (3.29) є задачею нелінійного програмування. Її множину допустимих розв'язків описують нерівності з негладкими функціями.

Псевдонормалізовані (adjusted) phi-функції та псевдонормалізовані квазі-phi-функції у (3.29) складаються, зазвичай, з max- та min- операцій диференційованих функцій. У результаті множина допустимих розв'язків  $W$  не є опуклою, що призводить до появи багатьох локальних екстремумів.

Однією з важливих особливостей допустимої множини (3.29) є те, що  $W = W_1 \cup \dots \cup W_s \cup \dots \cup W_\eta$ , де кожен підмножину  $W_s$  задають системою нерівностей з диференційованими функціями (див., наприклад, [131]).

Задачу (3.28), (3.29) можна звести до такої задачі оптимізації:

$$F(p^*, u^*, \tau^*) = \min \{F(p^{s*}, u^{s*}, \tau^{s*}), s = 1, 2, \dots, \eta\}, \quad (3.32)$$

де

$$F(p^{s*}, u^{s*}, \tau^{s*}) = \min_{(p, u, \tau) \in W_s} F(p, u, \tau). \quad (3.33)$$

Модель вимагає пошуку локальних екстремумів у всіх підмножинах та забезпечує загальний мінімум, якщо кожен підзадачу (3.33) можна розв'язати оптимально. Підзадачі (3.33) є задачами нелінійного програмування, їх можна безпосередньо розв'язати (принаймні теоретично) за допомогою глобальних NLP-солверів.

Згідно з особливостями  $\rho$ -функцій, квазі- $\rho$ -функцій та формами механічних обмежень, задіяних у (3.29), допустиму множину задачі (3.33) можна описати системою нерівностей з диференційованими функціями.

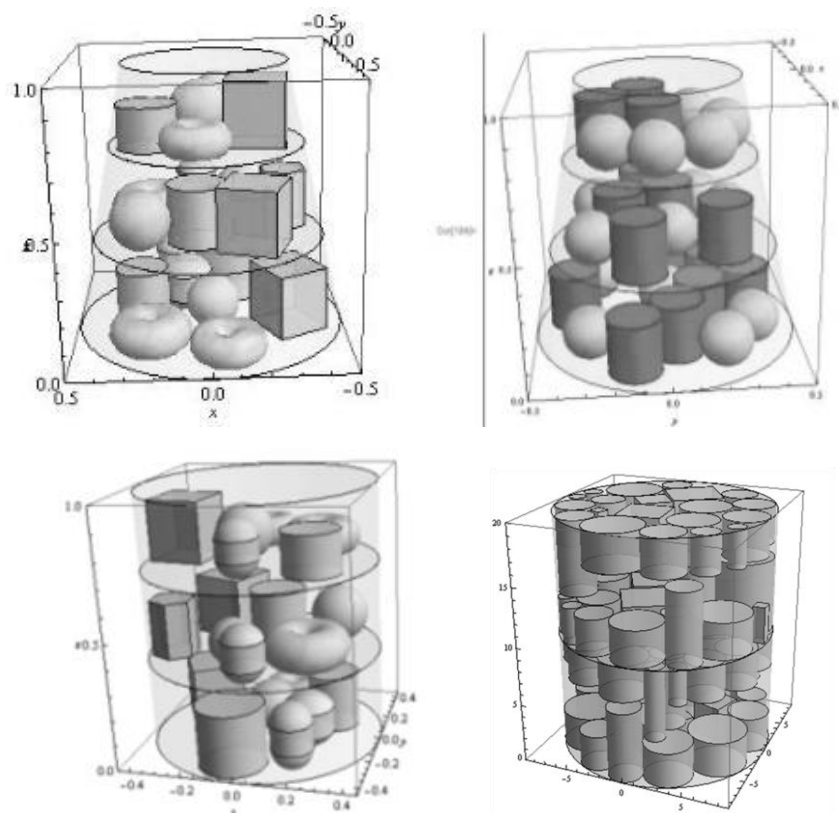
Для розв'язання кожної задачі нелінійного програмування (3.33) поєднуємо стратегію мультистарта з вибором початкових точок залежно від форми цільової функції та типів обмежень, використаних у (3.29).

З метою зменшення обчислювальних витрат (часу та пам'яті) використовуємо модифікацію алгоритму декомпозиції, запропоновану в [130]. Алгоритм дає змогу звести кожен задачу (3.33) великої розмірності до послідовності підзадач нелінійного програмування меншої розмірності.

На рис. 3.9 наведено кілька прикладів локальних оптимальних пакувань тривимірних об'єктів з різними цільовими функціями, умовами балансу та з урахуванням допустимих відстаней.

Цікавим напрямом подальших досліджень є вивчення багатокритеріального аналога задачі (3.28), (3.29). Насправді цільову функцію

(3.28) можна розглядати як лінійну комбінацію різних цілей  $F_k(p, u, \tau)$  з фіксованими вагами. Аналіз багатокритеріальної постановки цієї задачі є дійсно складним завданням, оскільки навіть для фіксованих ваг однокритеріальну задачу розв'язати важко. Наступний підрозділ присвячено розгляду саме цього питання.



**Рис. 3.9.** Локально-оптимальні пакування 3D об'єктів у зрізаному конусі і циліндрі з умовами балансу

### 3.6. Багатокритеріальна задача балансного компоунвання

У підрозділі розглянуто задачу балансного компоунвання 3D-об'єктів за такого формулювання: розмістити 3D-об'єкти в контейнері з урахуванням мінімально та максимально допустимих відстаней між об'єктами та умовами балансу так, щоб цільова функція досягала свого екстремального значення. При цьому задача балансного 3D-компоунвання формулюється у вигляді

багатокритеріальної задачі оптимізації – у подальшому багатокритеріальна задача балансного компонування (MBLP).

Для аналітичного опису обмежень неперетину, включення з урахуванням допустимих відстаней застосовують класи псевдонормалізованих  $\phi$ -функцій та псевдонормалізованих квазі- $\phi$ -функцій. Умови балансу (рівновага, моменти інерції та обмеження стабільності) формалізовано в [105]. Як контейнер розглядають коловий циліндр, параболоїд або зрізаний конус. Об'єктами розміщення є циліндри, сфери, тори, сферичні циліндри та прямі опуклі призми. Побудовано точну математичну модель задачі у вигляді NLP задачі.

**Постановка задачі MBLP.** Нехай  $\Omega = \{(x, y, z) \in \mathbb{R}^3 : G(x, y, z) \geq 0\}$  – контейнер висотою  $H$ . Розглянемо такі типи контейнерів: 1)  $\Omega \equiv \mathbf{C}$ ,  $\mathbf{C}$  – прямий коловий циліндр з основою радіусом  $R$ ,  $G(x, y, z) = \min\{-x^2 - y^2 + R^2, -z + H, z\}$ ; 2)  $\Omega \equiv \mathbf{\Lambda}$ ,  $\mathbf{\Lambda}$  – параболоїд обертання з основою радіусом  $R = \sqrt{H}$ ,  $G(x, y, z) = \min\{-z - x^2 - y^2 + H, z\}$ ; 3)  $\Omega \equiv \mathbf{E}$ ,  $\mathbf{E}$  – прямий коловий затуплений конус з нижньою і верхньою основами радіусами  $R_1$  та  $R_2 < R_1$  відповідно,

$$G(x, y, z) = \min\{-z - H(\sqrt{x^2 + y^2} - R_1)/(R_1 - R_2), -z + H, z\}.$$

Припустимо, що  $\Omega$  ділиться круглими стелажми  $S_k$ ,  $k = 1, 2, \dots, m+1$ , на підконтейнери  $\Omega^k$ ,  $k = 1, 2, \dots, m$ . Уважаємо, що  $S_1$  є базою  $\Omega$ . Між стелажми  $S_k$  і  $S_{k+1}$  відстань  $t_k$  задано.

Сімейство  $A = \{A_i, i \in I_n\}$ ,  $I_n = \{1, 2, \dots, n\}$ , включає в себе такі форми об'єктів: кулі  $S_i$  радіусом  $r_i$ ; прямі колові циліндри  $C_i$  радіусом  $r_i$  і висотою  $2h_i$ ; тори  $T_i$  з метричними характеристиками  $(r_i, h_i)$ , де  $r_i$  – відстань від центра твірного кола до осі обертання,  $2h_i$  є висотою  $T_i$ ,  $h_i$  – радіус твірного кола; сфероциліндри  $S_{Ci}$  з метричними характеристиками  $(l_i, r_i, h_i)$ , де  $l_i$  –

висоти кульових сегментів,  $r_i$  – радіус і  $2h_i$  – висота циліндра; прями правильні призми та кубоїди  $\mathbb{K}_i$  з метричними характеристиками  $(h_i, \tilde{v}_{il}, )$ , де  $2h_i$  – висота  $\mathbb{K}_i$ ,  $\tilde{v}_{il} = (x_{il}, y_{il})$ ,  $l=1, \dots, s_i$ , є вершинами основи  $\mathbb{K}_i$  (яка являє собою опуклий багатокутник  $K_i$ ),  $s_i$  – кількість вершин  $K_i$ .

Багатокритеріальна задача балансної компоновки (MBLP) полягає в такому.

Упакувати 3D-об'єкти  $A_i \in A$ ,  $i \in I_n = \{1, 2, \dots, n\}$ , всередині контейнера  $\Omega$  так, щоб вектор-функція цілі досягала екстремального значення з урахуванням геометричних обмежень та умов балансу.

Обмеження розміщення в задачі MBLP генеруються умовами неперетину об'єктів  $A_i, A_j$ ,  $i > j \in I_n$ , які мають бути розміщені всередині контейнера  $\Omega$ , і включення об'єкта  $A_i$  в контейнер  $\Omega$ ,  $i \in I_n$ . Крім того, мінімальна  $\rho_{ij}^-$  і максимальна  $\rho_{ij}^+ \geq \rho_{ij}^-$  допустимі відстані між об'єктами  $A_i, A_j$ ,  $i > j \in I_n$ , можуть бути задані. До того ж, може бути визначена мінімально допустима відстань  $\rho_i^-$  між об'єктом  $A_i \in A$ ,  $i \in I_n$ , і боковою поверхнею контейнера  $\Omega$ . Без обмеження загальності встановлюємо  $\rho_{ij}^- = 0$  (або  $\rho_{ij}^+ = \varpi$ ), якщо мінімальну (або максимальну) допустиму відстань між об'єктами  $A_i$  та  $A_j$  не задано,  $i > j \in I_n$ . Тут  $\varpi$  – задане досить велике число. Зокрема, умова  $\rho_{ij}^+ = \rho_{ij}^-$  забезпечує розташування об'єктів  $A_i$  та  $A_j$  на точній відстані. Також встановлюємо  $\rho_i^- = 0$ , якщо мінімальну допустиму відстань між об'єктом  $A_i$  і боковою поверхнею контейнера  $\Omega$  не задано.

Обмеження розміщення в задачі MBLP може бути записано так:  
 $\rho_{ij}^- \leq \text{dist}(A_i, A_j) \leq \rho_{ij}^+$ ,  $i > j \in I_n$ , і  $\text{dist}(A_i, \Omega^*) \geq \rho_i^-$ ,  $i=1, \dots, n$ , де  
 $\Omega^* = \mathbb{R}^3 \setminus \text{int } \Omega$ .

Для аналітичного опису зазначених вище обмежень розміщення використовуємо метод  $\phi$ -функцій (див. п. 1.4).

Розглянемо обмеження механічних характеристик системи  $\Omega_A$ .

Обмеження рівноваги визначаються такою системою нерівностей:

$$\begin{aligned}\mu_{11}(u) &= \min\{-(x_s(u) - x_e) + \Delta x_e, (x_s(u) - x_e) + \Delta x_e\} \geq 0 \\ \mu_{12}(u) &= \min\{-(y_s(u) - y_e) + \Delta y_e, (y_s(u) - y_e) + \Delta y_e\} \geq 0, \\ \mu_{13}(u) &= \min\{-(z_s(u) - z_e) + \Delta z_e, (z_s(u) - z_e) + \Delta z_e\} \geq 0,\end{aligned}$$

де  $(x_e, y_e, z_e)$  – очікувана позиція  $O_s$ ,  $(\Delta x_e, \Delta y_e, \Delta z_e)$  є допустимими відхиленнями від точки  $(x_e, y_e, z_e)$ .

Обмеження моментів інерції визначаються так:

$$\begin{aligned}\mu_{21}(u) &= -J_X(u) + \Delta J_X \geq 0, \\ \mu_{22}(u) &= -J_Y(u) + \Delta J_Y \geq 0, \\ \mu_{23}(u) &= -J_Z(u) + \Delta J_Z \geq 0,\end{aligned}$$

де  $J_X(u), J_Y(u), J_Z(u)$  – моменти інерції системи  $\Omega_A$  відносно осей системи координат  $O_sXYZ$ ,  $\Delta J_X, \Delta J_Y, \Delta J_Z$  є допустимими значеннями для  $J_X(u), J_Y(u), J_Z(u)$ , де

$$J_X(u) = J_{x_0} + \sum_{i=1}^n (J_{x_i} \cos^2 \theta_i + J_{y_i} \sin^2 \theta_i) + \sum_{i=1}^n (y_i^2 + z_i^2) m_i - M(y_s^2 + z_s^2),$$

$$J_Y(u) = J_{y_0} + \sum_{i=1}^n (J_{x_i} \sin^2 \theta_i + J_{y_i} \cos^2 \theta_i) + \sum_{i=1}^n (x_i^2 + z_i^2) m_i - M(x_s^2 + z_s^2)$$

$$J_Z(u) = \sum_{i=0}^n J_{z_i} + \sum_{i=1}^n (y_i^2 + z_i^2) m_i - M(x_s^2 + y_s^2),$$

$J_{x_0}, J_{y_0}, J_{z_0}$  – моменти інерції контейнера  $\Omega$  відносно осей системи координат  $Oxyz$ ,  $J_{x_i}, J_{y_i}, J_{z_i}$ ,  $i \in I_n$ , – моменти інерції об'єкта  $A_i$  відносно осей системи координат  $O_i x_i y_i z_i$ .

Обмеження стабільності визначаються такою системою нерівностей:

$$\mu_{31}(u) = \min\{-J_{XY}(u) + \Delta J_{XY}, J_{XY}(u) + \Delta J_{XY}\} \geq 0,$$

$$\mu_{32}(u) = \min\{-J_{YZ}(u) + \Delta J_{YZ}, J_{YZ}(u) + \Delta J_{YZ}\} \geq 0,$$

$$\mu_{33}(u) = \min\{-J_{XZ}(u) + \Delta J_{XZ}, J_{XZ}(u) + \Delta J_{XZ}\} \geq 0,$$

де  $J_{XY}(u), J_{YZ}(u), J_{XZ}(u)$  є моменти інерції системи  $\Omega_A$  відносно осей системи координат  $O_s XYZ$ ,  $\Delta J_{XY}, \Delta J_{YZ}, \Delta J_{XZ}$  є допустимими значеннями для  $J_{XY}(u), J_{YZ}(u), J_{XZ}(u)$ , відповідно,

$$J_{XY}(u) = \frac{1}{2} \sum_{i=1}^n (J_{x_i} - J_{y_i}) \sin 2\theta_i + \sum_{i=1}^n x_i y_i m_i - M x_s y_s,$$

$$J_{YZ}(u) = \sum_{i=1}^n y_i z_i m_i - M y_s z_s, \quad J_{XZ}(u) = \sum_{i=1}^n x_i z_i m_i - M x_s z_s.$$

Умови балансу задачі MBLP визначаємо як систему нерівностей  $\mu_1(u) \geq 0, \mu_2(u) \geq 0, \mu_3(u) \geq 0$ , де

$$\mu_1(u) = \min\{\mu_{11}(u), \mu_{12}(u), \mu_{13}(u)\}, \quad (3.34)$$

$$\mu_2(u) = \min\{\mu_{21}(u), \mu_{22}(u), \mu_{23}(u)\}, \quad (3.35)$$

$$\mu_3(u) = \min\{\mu_{31}(u), \mu_{32}(u), \mu_{33}(u)\}. \quad (3.36)$$

Тут  $O_s = (x_s, y_s, z_s)$  є центром мас системи  $\Omega_A$ ,  $x_s(u) = \frac{1}{M} \sum_{i=1}^n m_i x_i$ ,

$$y_s(u) = \frac{1}{M} \sum_{i=1}^n m_i y_i, \quad z_s(u) = \frac{1}{M} \sum_{i=1}^n m_i z_i, \quad M = \sum_{i=0}^n m_i \text{ – маса системи } \Omega_A.$$

**Математична модель задачі MBLP.** Математичну модель задачі MBLP можна навести у формі

$$\text{extr}F(p, u), (p, u) \in W, \quad (3.37)$$

$$W = \{(p, u) \in \mathbb{R}^\xi : \Upsilon(p, u) \geq 0, \mu(p, u) \geq 0, \zeta \geq 0\}, \quad (3.38)$$

де  $F(p, u) = (F_1(p, u), F_2(p, u), \dots, F_k(p, u))$ ,  $\Upsilon(p, u)$  описує обмеження розміщення,  $\Upsilon(p, u) = \min\{\Upsilon_1(u), \Upsilon_2(p, u)\}$ ,  $\Upsilon_1(u)$  відповідає за обмеження неперетину,  $\Upsilon_2(p, u)$  відповідає за обмеження включення,  $\mu(p, u) = \min\{\mu_s(p, u), s \in U_t\}$  відповідає за умови балансу,  $U_t \in P(U)$ ,  $P(U)$  – потужність множини  $U = \{1, 2, 3\}$ , функції  $\mu_1(p, u), \mu_2(p, u), \mu_3(p, u)$  задано співвідношеннями (3.34)–(3.36),  $\zeta \geq 0$  є системою додаткових обмежень метричних характеристик контейнера  $\Omega$  та параметрів розміщення об’єктів. Якщо  $s = \emptyset$ , тобто умови балансу не беруть участь у (3.38), тоді цільова функція  $F(p, u)$  відповідає механічним характеристикам системи  $\Omega_A$ .

Залежно від різних комбінацій цільових функцій  $F_1(p, u), F_2(p, u), \dots, F_k(p, u)$  можуть бути сформовані різні варіанти математичної моделі (3.37), (3.38). Найчастіше зустрічаються цільові функції, розглянуті у відповідних публікаціях [105, 120, 123, 132–134], а саме:

1) розмір контейнера  $\Omega$ ; 2) відхилення центра мас системи  $\Omega_A$  від заданої точки; 3) моменти інерції системи  $\Omega_A$ .

Розглянемо деякі реалізації моделі (3.37), (3.38):

- $F(p, u) = p, (p, u) \in W \subset \mathbb{R}^\xi,$

$$W = \{(p, u) \in \mathbb{R}^\xi : \Upsilon_1(u) \geq 0, \Upsilon_2(p, u) \geq 0, \mu(p, u) \geq 0, \zeta \geq 0\};$$

$$\bullet F(u) = d, (p, u) \in W \subset \mathbb{R}^\xi,$$

$$d = (x_s(u) - x_e)^2 + (y_s(u) - y_e)^2 + (z_s(u) - z_e)^2,$$

$$W = \{(p, u) \in \mathbb{R}^\xi : \Upsilon_1(u) \geq 0, \Upsilon_2(p, u) \geq 0, \mu_2(p, u) \geq 0, \mu_3(p, u) \geq 0, \zeta \geq 0\};$$

$$\bullet F(p, u) = (F_1(p, u) = p, F_2(p, u) = d), (p, u) \in W \subset \mathbb{R}^\xi,$$

$$W = \{(p, u) \in \mathbb{R}^\xi : \Upsilon_1(u) \geq 0, \Upsilon_2(p, u) \geq 0, \mu_2(p, u) \geq 0, \mu_3(p, u) \geq 0, \zeta \geq 0\};$$

$$\bullet F(p, u) = (F_1(p, u) = J_X(p, u), F_2(p, u) = J_Y(p, u), F_3(p, u) = J_Z(p, u))$$

$$(p, u) \in W \subset \mathbb{R}^\xi,$$

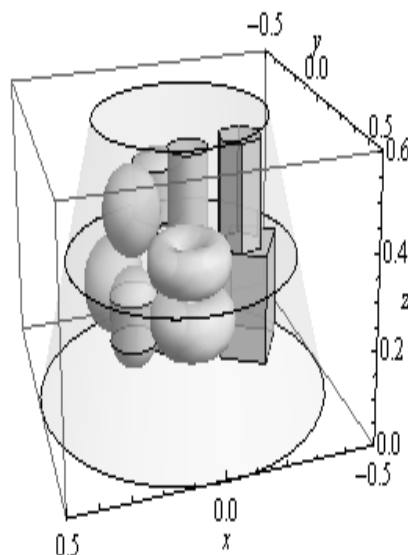
$$W = \{(p, u) \in \mathbb{R}^\xi : \Upsilon_1(u) \geq 0, \Upsilon_2(p, u) \geq 0, \mu_1(p, u) \geq 0, \mu_3(p, u) \geq 0, \zeta \geq 0\}.$$

Результати обчислювальних експериментів:

Приклад 3.3. Нехай  $\Omega \equiv \mathbf{E}$ ,  $m = 2$ ,  $H = 0.6, R_1 = 0.5, R_3 = 0.3$ ,  
 $A = \{S_1, S_2, C_3, C_4, T_5, T_6, S_{C7}, S_{C8}, K_9, K_{10}\}$ ,  $A_-^1 = \{S_1, C_3, T_5, S_{C7}, K_9\}$ ,  
 $A_+^2 = \{S_2, C_4, T_6, S_{C8}, K_{10}\}$ ,  $\rho_{ij}^- = 0.03$ ,  $i < j \in I_{10}$ ,  $\rho_{39}^+ = 0.1$ ,  $\rho_{26}^+ = 0.08$ ,  
 $(x_e, y_e, z_e) = (0, 0, 0.275)$ ,  $t_1 = 0.3$ ,  $n = 10$ ,

$\{z_i, i = 1, \dots, 10\} = \{0.19, 0.4, 0.19, 0.41, 0.24, 0.35, 0.19, 0.39, 0.18, 0.42\}$ ,  
 $\{m_i, i = 1, \dots, 10\} = \{27.8764, 20.944, 34.5575, 16.9332, 28.4245, 22.2066, 17.2159,$   
 $19.2265, 38.4, 19.9532\}$ ,  $r_1 = 0.11$ ,  $r_2 = 0.1$ ,  $r_3 = 0.1$ ,  $h_3 = 0.11$ ,  $r_4 = 0.07$ ,  
 $h_4 = 0.11$ ,  $r_5 = 0.08$ ,  $h_5 = 0.06$ ,  $r_6 = 0.09$ ,  $h_6 = 0.05$ ,  $r_7 = 0.08$ ,  $h_7 = 0.05$ ,  
 $l_7 = 0.06$ ,  $h_8 = 0.06$ ,  $l_8 = 0.03$ ,  $s_9 = 4$ ,  $h_9 = 0.12$ ,  $\tilde{v}_{91} = (0.08, 0.1)$ ,  $\tilde{v}_{92} = (0.08,$   
 $0.1)$ ,  $\tilde{v}_{93} = (-0.08, -0.1)$ ,  $\tilde{v}_{94} = (-0.08, 0.1)$ ,  $s_{10} = 6$ ,  $h_{10} = 0.12$ ,  
 $\tilde{v}_{(10)1} = (0.04, 0.07)$ ,  $\tilde{v}_{(10)2} = (0.08, 0)$ ,  $\tilde{v}_{(10)3} = (0.04, -0.07)$ ,  $\tilde{v}_{(10)4} = (-0.04, -0.07)$ ,  
 $\tilde{v}_{(10)5} = (-0.08, 0)$ ,  $\tilde{v}_{(10)6} = (-0.04, 0.07)$ .

Локальними оптимальними розв'язками, знайденими NLP-солвером CAS Math 9 (рис. 3.10), є  $F(u^*, u'^*) = 1.12726 \times 10^{-6}$ .

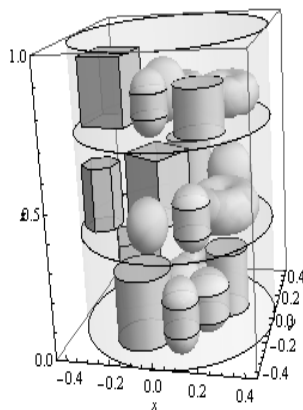


**Рис. 3.10.** Локальне оптимальне розміщення 3D-об'єктів у прикладі 3.3

Приклад 3.4. Нехай  $\Omega \equiv \mathbf{C}$ ,  $m=3$ ,  $H=1$ ,  $R=0.45$ ,  $t_2=0.35$ ,  $n=20$ ,  
 $A = \{S_i, i=1, \dots, 4, \quad C_i, i=5, \dots, 8, \quad T_i, i=9 \dots 12, \quad S_{C_i}, i=13, \dots, 16,$   
 $K_i, i=17, \dots, 20\}$ ,  $A_+^1 = \{S_1, C_5, C_6, T_9, S_{C14}, P_{17}\}$ ,  $A_+^2 = \{S_2, S_3, C_7, T_{10},$   
 $S_{C15}, P_{18}, K_{20}\}$ ,  $A_+^3 = \{S_4, C_8, T_{11}, T_{12}, S_{C16}, P_{19}\}$ ,  $U_t = \emptyset$ ,  $\rho_{ij}^- = 0.02$ ,  
 $i < j = 1, \dots, 20$ ,  $(x_e, y_e, z_e) = (0, 0, 0.5)$ ,  $\{z_i, i=1, \dots, 20\} = \{0.1, 0.44, 0.46, 0.81,$   
 $0.11, 0.12, 0.46, 0.78, 0.06, 0.425, 0.76, 0.77, 0.11, 0.13, 0.46, 0.81, 0.12, 0.47, 0.82,$   
 $0.46\}$ ,  $\{m_i, i=1, \dots, 20\} = \{20.944, 15.2681, 27.8764, 34.5575, 63.7115, 41.8146,$   
 $30.4106, 28.4245, 49.9649, 24.8714, 38.6888, 26.2637, 20.7764, 17.2159, 16.8756,$   
 $52.8, 52.8, 52.8, 23.1489\}$ ,  $r_1=0.1$ ,  $r_2=0.09$ ,  $r_3=0.11$ ,  $r_4=0.11$ ,  $r_5=0.1$ ,  
 $h_5=0.11$ ,  $h_6=0.12$ ,  $r_7=0.11$ ,  $r_8=0.11$ ,  $h_8=0.08$ ,  $r_9=0.08$ ,  $h_9=0.07$ ,  
 $r_{10}=0.09$ ,  $h_{10}=0.075$ ,  $r_{11}=0.07$ ,  $h_{11}=0.06$ ,  $r_{12}=0.08$ ,  $h_{12}=0.07$ ,  
 $r_{13}=0.1$ ,  $h_{13}=0.05$ ,  $l_{13}=0.07$ ,  $r_{14}=0.05$ ,  $h_{14}=0.05$ ,  $l_{14}=0.08$ ,  $r_{15}=0.08$   
,  $h_{15}=0.05$ ,  $l_{15}=0.06$ ,  $r_{16}=0.08$ ,  $h_{16}=0.04$ ,  $l_{16}=0.07$ ,  $s_i=4$ ,  
 $\tilde{v}_{i1} = (-0.11, -0.1)$ ,  $\tilde{v}_{i2} = (0.11, -0.1)$ ,  $\tilde{v}_{i3} = (0.11, 0.1)$ ,  $\tilde{v}_{i4} = (-0.11, 0.1)$ ,

$h_i = 0.12, i = 17, 18, 19, s_{20} = 6, \tilde{v}_{(20)1} = (0.045, 0.078), \tilde{v}_{(20)2} = (0.09, 0), \tilde{v}_{(20)3} = (0.045, -0.078),$   
 $\tilde{v}_{(20)4} = (-0.045, -0.078), \tilde{v}_{(20)5} = (-0.09, 0),$   
 $\tilde{v}_{(20)6} = (-0.045, 0.078), h_{20} = 0.11.$

Локальними оптимальними розв'язками, знайденими NLP-солвером CAS Math 9,  $\epsilon F(u^*, u'^*) = 0.001911$  (рис. 3.11).

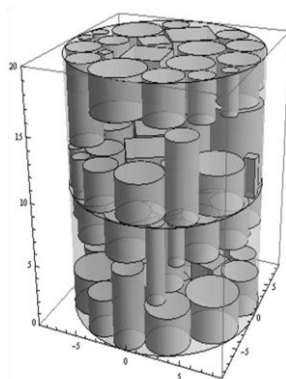


**Рис. 3.11.** Локальне оптимальне розміщення 3D-об'єктів у прикладі 3.4

Приклад 3.5. Нехай  $\Omega \equiv \mathbf{C}, m = 2, H = 20, R = 8.8, t_1 = 10, n = 80,$   
 $A = \{C_i, i = 1, \dots, 64, K_j, i = 65, \dots, 80\}, A_+^1 = \{C_i, i = 1, \dots, 16, K_j, i = 65, \dots, 68\},$   
 $A_-^1 = \{C_i, i = 17, \dots, 32, K_j, i = 69, \dots, 72\}, A_-^2 = \{C_i, i = 49, \dots, 64, K_j, i = 77,$   
 $\dots, 80\}, A_+^2 = \{C_i, i = 33, \dots, 48, K_j, i = 73, \dots, 76\}, \{r_i, i = 1, \dots, 64\} = \{2.0, 2.4, 0.8,$   
 $1.1, 1.3, 0.7, 0.7, 1.5, 2.4, 1.8, 1.5, 1.7, 1.7, 1.4, 1.6, 2.1, 2.0, 2.4, 0.8, 1.1, 1.3, 0.7,$   
 $0.7, 1.5, 2.4, 1.8, 1.5, 1.7, 1.7, 1.4, 1.6, 2.1, 2.0, 2.4, 0.8, 1.1, 1.3, 0.7, 0.7, 1.5, 2.4,$   
 $1.8, 1.5, 1.7, 1.7, 1.4, 1.6, 2.1, 2.0, 2.4, 0.8, 1.1, 1.3, 0.7, 0.7, 1.5, 2.4, 1.8, 1.5, 1.7,$   
 $1.7, 1.4, 1.6, 2.1\}, s_j = 4, i = 65, \dots, 80, \tilde{v}_{i1} = (-1.8, -1.8), \tilde{v}_{i2} = (1.8, -1.8),$   
 $\tilde{v}_{i3} = (1.8, 1.8), \tilde{v}_{i4} = (-1.8, 1.8), i = 65, 69, 73, 77, \tilde{v}_{i1} = (-0.5, -0.5),$   
 $\tilde{v}_{i2} = (0.5, -0.5), \tilde{v}_{i3} = (0.5, 0.5), \tilde{v}_{i4} = (-0.5, 0.5), i = 66, 70, 74, 78,$   
 $\tilde{v}_{i1} = (-2.1, -2.1), \tilde{v}_{i2} = (2.1, -2.1), \tilde{v}_{i3} = (2.1, 2.1), \tilde{v}_{i4} = (-2.1, 2.1),$   
 $i = 67, 71, 75, 79, \tilde{v}_{i1} = (-1.3, -1.3), \tilde{v}_{i2} = (1.3, -1.3), \tilde{v}_{i3} = (1.3, 1.3),$

$\tilde{v}_{i4} = (-1.3, 1.3)$ ,  $i = 68, 72, 76, 80$ ,  $\{h_i, i = 1, \dots, 62\} = \{1.5, 1.5, 1.5, 1.5, 1.5, 3.0,$   
 $1.5, 1.5, 1.5, 3.0, 1.5, 3.0, 1.5, 3.0, 3.0, 1.5, 1.5, 1.5, 3.0, 1.5, 3.0, 1.5, 1.5, 3.0, 1.5,$   
 $1.5, 3.0, 1.5, 1.5, 3.0, 1.5, 1.5, 1.5, 3.0, 1.5, 1.5, 1.5, 3.0, 1.5, 1.5, 1.5, 3.0, 1.5, 1.5,$   
 $1.5, 3.0, 3.0, 1.5, 1.5, 1.5, 3.0, 1.5, 3.0, 1.5, 1.5, 3.0, 1.5, 1.5, 3.0, 1.5, 1.5, 3.0, \{$   
 $h_i, i = 63, \dots, 80\} = \{1.5\}$ ,  $\{m_i, i = 1, \dots, 80\} = \{86, 72, 81, 54, 29, 94, 92, 41, 57, 77,$   
 $40, 67, 31, 47, 39, 61, 73, 83, 11, 20, 86, 72, 81, 54, 29, 94, 92, 41, 57, 77, 40, 67,$   
 $31, 47, 39, 61, 73, 83, 11, 20, 86, 72, 81, 54, 29, 94, 92, 41, 57, 77, 40, 67, 31, 47,$   
 $39, 61, 73, 83, 11, 20, 86, 72, 81, 54, 29, 94, 92, 41, 57, 77, 40, 67, 31, 47, 39, 61,$   
 $73, 83, 11, 20\}$ .

Локальними оптимальними розв'язками, знайденими ПРОРТ, є  $F(u^*, u'^*) = 0.000000$  (рис. 3.12).



**Рис. 3.12.** Локальне оптимальне розміщення 3D-об'єктів у прикладі 3.5  
 Деякі результати, наведені в розділі, опубліковано в працях [41, 43, 46, 53, 55, 56, 58].

## РОЗДІЛ 4. ЗАДАЧІ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ НА ЗАЛІЗНИЧНОМУ ТЕРМІНАЛІ

Досліджено підвищення ефективності роботи залізничних терміналів. Сформульовано проблеми побудови оптимального розкладу обслуговування потягів і оптимізації розміщення контейнерів на залізничних платформах та в зоні зберігання на залізничному терміналі. Наведено підходи до розв'язання задачі на різних рівнях деталізації. Побудовано комбінаторні оптимізаційні моделі задачі на двох рівнях, проаналізовано їхні властивості, розглянуто приклади.

### 4.1. Контейнерні операції на залізничному терміналі

Контейнерні операції на залізничних терміналах мають велике значення в сучасній логістиці [81, 153]. Більшість із них полягає в обробленні контейнерів, що мають форму паралелепіпедів, на залізничних перевантажувальних станціях (терміналах) [83–85]. З огляду на це, однією з актуальних проблем є оптимізація роботи залізничного терміналу під час оброблення вантажних потягів на залізничному перевалочному майданчику. Можна оптимізувати такі операції: планування процесів (прибуття, відправлення, завантаження та розвантаження) потягів і розміщення контейнерів з цих потягів на залізничних платформах та на відкритому складському майданчику (у зоні зберігання). У деяких ситуаціях формулюються додаткові вимоги до балансування контейнерів та об'єктів усередині них. Дослідженню проблем цього класу присвячено чимало наукових публікацій (див., наприклад, [81–85, 153–155] та посилання в них).

Обслуговування вантажних потягів включає такі процеси [83–85]: прибуття потягу на залізничний термінал, завантаження та розвантаження контейнерів із потягу та відправлення потягу із залізничного терміналу. Потяг може повернутися до станції, щоб завершити завантаження. Оброблення

контейнерів полягає у перенесенні контейнера безпосередньо з одного потягу до іншого, перенесенні контейнера з потягу в зону зберігання, перенесенні контейнера із зони зберігання до потягу.

Сучасний залізничний термінал складається з декількох основних елементів [85]:

- 1) платформа з певною кількістю паралельних колій; на цих коліях розташовані потяги;
- 2) зона для зберігання контейнерів, яка розташована паралельно коліям;
- 3) козлові крани, які переміщують контейнери безпосередньо між потягами та між потягами та зоною зберігання.

Є залізничні колії, на які потяги прибувають і з якими залишають залізничний термінал, або чекають обслуговування перед залізничним терміналом і позаду нього. Крім того, залізничні термінали можуть бути обладнані автомобільними під'їзними шляхами. Такий залізничний термінал забезпечує мультимодальні транспортні послуги. Схему типового залізничного терміналу наведено на рис. 4.1 [85].

Залізничний термінал призначений для того, щоб з потягів перевантажувати контейнери на їхньому шляху. На станцію прибуває потяг з контейнерами для інших потягів (потяг-джерело). Контейнери перевантажують на інші (цільові) потяги безпосередньо на станції або в зону зберігання для подальшого завантаження. Якщо в зоні зберігання є контейнери для цільового потягу, то їх слід перенести на вільні платформи цього потягу. Після здійснення всіх операцій з перевантаження контейнерів потяг залишає залізничний термінал.

У статті [85] наведено огляд проблем планування робіт на залізничному терміналі. Проаналізовано актуальні проблеми, рішення яких дає змогу підвищити ефективність перевантажувальних операцій на залізничному терміналі.

Загальна проблема планування роботи залізничного термінала, сформульована в [85], полягає в плануванні руху вантажних потягів та операцій перевантаження контейнерів. Через складність загальної проблеми запропоновано такі рівні для її вирішення [84, 85]:

I. Розклад обслуговування слотів потягів (Transshipment Yards Scheduling Problem, TYSP).

II. Призначення кожного потягу на залізничну колію.

III. Ухвалення рішення про позиції контейнерів на потягах.

IV. Призначення переміщень контейнерів кранами.

V. Визначення послідовності переміщення контейнерів кранами.

У праці [85] запропоновано вирішення проблеми на рівні I (TYSP). Розроблено підхід до планування руху потягів, що прибувають на залізничний термінал. Відповідно до підходу потяги призначаються в сервісні слоти для одночасного обслуговування. Побудовано математичну модель задачі TYSP на основі булевих змінних, для аналізу моделі та розв'язання задачі використано евристичний алгоритм.

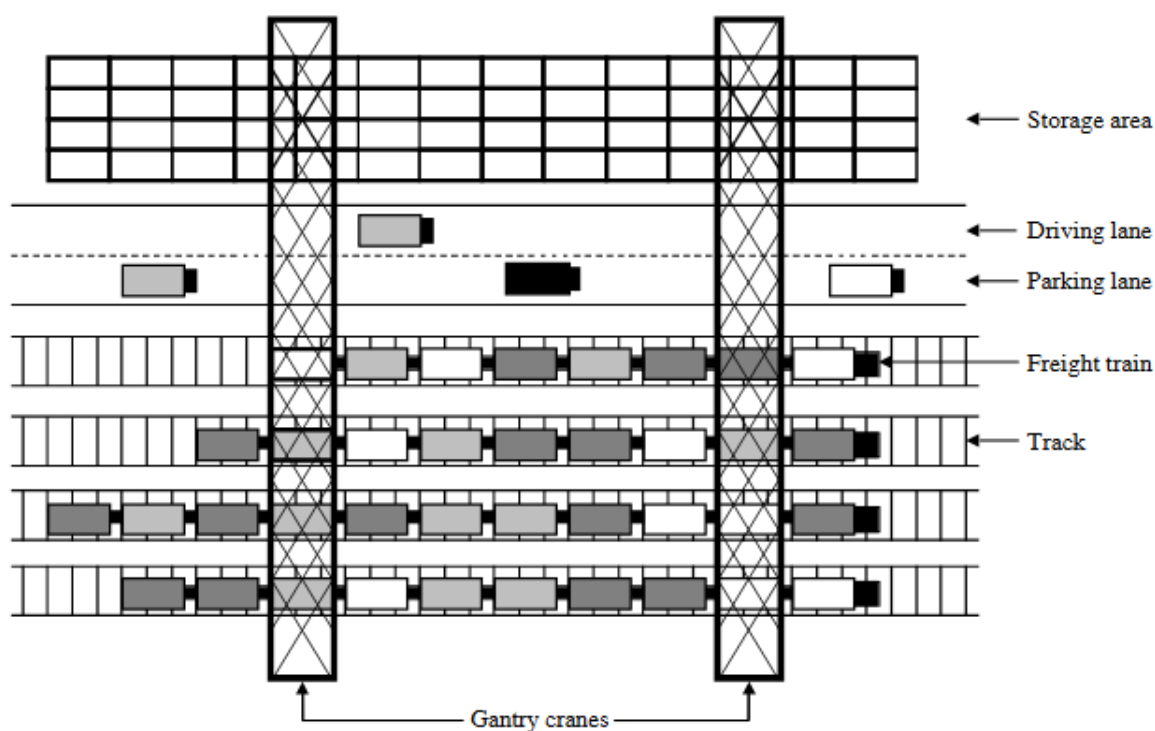


Рис. 4.1. Схема типового залізничного термінала

#### 4.2. Задача складання розкладу руху вантажних потягів та оброблення вантажів на залізничному терміналі

Опишемо задачу TYSP в оригінальній постановці, наведеній у [85]. Задано множину потягів  $I = \{1, 2, \dots, N\}$ . Кожен потяг везе контейнери, які потрібно перевантажити на один чи декілька інших (цільових) потягів, та має отримати контейнери від одного чи декількох потягів. Матриця  $A_{N \times N}$  задає кількість контейнерів, які з потягу  $i$  потрібно перевантажити на потяг  $j$ ,  $i, j = 1, 2, \dots, N$ . Перевантаження контейнерів з одного потягу на інший здійснюється на залізничному терміналі за допомогою козлових кранів. Залізничний термінал має  $G$  колій ( $G$  значно менше  $N$ ), час роботи терміналу розбито на проміжки – сервісні слоти (service slot), протягом одного слота обслуговується максимум  $G$  потягів. Необхідно знайти оптимальний за вартістю розклад прибуття потягів та перевантаження контейнерів на залізничному терміналі.

У процесі обслуговування потягів на залізничному терміналі можливі такі ситуації:

- повторне відвідування (revisit), коли після обслуговування потяг повинен повторно заходити на залізничний термінал для завантаження контейнерів, які надійшли пізніше;
- розділені переміщення контейнерів (split moves), коли потяг  $i$ , що перевозить контейнери для потягу  $j$ , обслуговується в сервісному слоті  $t$  раніше сервісного слота  $t'$  потягу  $j$ . Тоді контейнери з потягу  $i$  тимчасово переміщуються в зону зберігання (storage area), звідки потім його перевантажують на потяг  $j$  в сервісному слоті  $t'$ .

У праці [85] описано розв'язання лише першого рівня задачі – формування сервісних слотів, тобто Transshipment Yards Scheduling Problem, TYSP. На цьому рівні розв'язком задачі є розбиття множини потягів на сервісні слоти, тобто розклад обслуговування потягів. При цьому для кожного

потягу  $i$  береться до уваги найбільш ранній  $e_i$  та найбільш пізній  $l_i$  допустимі слоти прибуття на залізничний термінал.

Розглянемо другий рівень деталізації задачі TYSP [85]. Під час формування сервісних слотів для обслуговування потягів приймається рішення щодо призначення потягів на певні колії. Будемо ґрунтуватися на таких міркуваннях.

1. Якщо вихідний і цільовий потяги обслуговуються в одному і тому самому сервісному слоті, доцільно призначити їх на якомога ближчі одна до одної колії.

2. Якщо вихідний і цільовий потяги обслуговуються в різних сервісних слотах, важливо призначити колії вихідному і цільовому потягам відповідно до того, щоб відстань, пройдена козловим краном під час обслуговування цих потягів, була мінімальною. У цьому випадку кран має спочатку перемістити контейнери з вихідного потягу до зони зберігання, а потім, у наступному слоті, із зони зберігання на цільовий потяг, отже, вихідний та цільовий потяги слід призначити на колії, розміщені якомога ближче до зони зберігання.

Врахування призначення потягів на колії дає змогу знизити витрати на обслуговування потягів на залізничному терміналі за рахунок скорочення відстані, яку долає козловий кран під час перевантаження контейнерів.

Розглянемо задачу знаходження оптимального розкладу прибуття вантажних потягів та перевантаження контейнерів на залізничному терміналі з призначенням потягів на колії на підставі відомих параметрів: кількості колій; кількості контейнерів для перевантаження з кожного потягу на інші; обмежень на найбільш ранній  $e_i$  та найбільш пізній  $l_i$  допустимі слоти прибуття потягу  $i$ ; вартості одиниці відстані переміщення контейнера козловим краном.

Побудуємо математичну модель задачі. Зобразимо сервісний слот  $t = 1, 2, \dots, T$  за допомогою кортежу  $K^t = (k_1^t, k_2^t, \dots, k_g^t, \dots, k_G^t)$ , що містить номери всіх потягів, які обслуговуються в слоті; порядок потягів відповідає їх призначенню на колії. Тут  $k_g^t \in I$ ,  $g = 1, 2, \dots, G$  – потяг, який обслуговується в слоті  $t$  і призначений на колію  $g$ .

Розклад прибуття потягів та оброблення вантажів на залізничному терміналі зображують у вигляді послідовності кортежів відповідно до слотів  $K^1, K^2, \dots, K^T$ , що визначає змінні задачі:  $X = (K^1, K^2, \dots, K^T) \in \mathbb{R}^{G \cdot T}$ .

Наведемо обмеження задачі у формальному вигляді:

$$e_{\kappa_g^t} \leq t \leq l_{\kappa_g^t}, t = 1, 2, \dots, T, g = 1, 2, \dots, G, \quad (4.1)$$

$$\kappa_p^t, \kappa_q^{t'} : t \leq t' + y_{\kappa_q^{t'}} \cdot M, t = 1, 2, \dots, T, t' = t + 1, \dots, T, p, q = 1, 2, \dots, G, \quad (4.2)$$

де  $T$  – кількість сервісних слотів,  $t$  – номер слота;  $G$  – кількість колій на залізничному терміналі;  $e_i$  та  $l_i$  – найбільш ранній та найбільш пізній допустимі слоти прибуття потягу  $i$ ;  $M$  – достатньо велике ціле число;  $y_i$  – булева змінна, що дорівнює 1, якщо потяг  $i$  повторно відвідує залізничний термінал, тобто входить до двох сервісних слотів  $t \neq t'$ :

$$y_i = \begin{cases} 1, & \text{якщо } \exists t, t', g, g' : t \neq t', \kappa_g^t = \kappa_{g'}^{t'} = i, \\ 0, & \text{в протилежному випадку.} \end{cases}$$

Обмеження (4.1) гарантує, що кожний потяг обслуговуватиметься не раніше слота  $e_i$  та не пізніше слота  $l_i$ .

Умова (4.2) забезпечує прибуття потягу-джерела  $\kappa_p^t$  перед цільовим потягом  $\kappa_q^{t'}$ ; крім того, якщо цільовий потяг відвідає термінал повторно (revisit), то умова завжди виконується завдяки доданку  $y_{\kappa_q^{t'}} \cdot M$ .

Уведемо позначення:

$$z_{tt'} = \begin{cases} 1, & \text{if } t < t' \\ 0, & \text{if } t = t' \end{cases} - \text{булева змінна, що дорівнює 1, якщо сервісні слоти } t \text{ та}$$

$t'$  є різними, та 0 – якщо однаковими;

$C_{pq}$  – вартість перевантаження козловим краном одного контейнера з потягу, розміщеного на колії  $p$ , на потяг, розміщений на колії  $q$ , в межах одного слота;

$C_{pq}^*$  – вартість перевантаження козовим краном одного контейнера з потягу, розміщеного на колії  $p$ , на потяг, розміщений на колії  $q$ , в різних слотах (вартість роздільного руху);

$C^r$  – вартість повторного відвідування залізничного терміналу потягом.

Величини  $C_{pq}$  та  $C_{pq}^*$  не залежать від слота і можуть бути заздалегідь визначені:

$$C_{pq} = c_{pq}, \quad (4.3)$$

$$C_{pq}^* = c_{p0} + c_{0q}, \quad (4.4)$$

де  $c_{pq}$  – фіксована вартість переміщення контейнера між коліями  $p$  та  $q$ .

Зона зберігання позначається як колія 0. У разі безпосереднього перевантаження контейнера з потягу на потяг вартість  $C_{pq}$  дорівнює вартості безпосереднього переміщення контейнера  $c_{pq}$ . У разі обслуговування потягів у різних слотах контейнер перевантажуватимуть через зону зберігання:

$$C_{pq}^* = c_{p0} + c_{0q}.$$

Значення  $c_{pq}$  залежать від особливостей конкретного залізничного терміналу. Вважатимемо, що вартість переміщення одного контейнера визначається виключно відстанню між залізничними коліями, а сусідні колії знаходяться на однаковій відстані, тому

$$c_{pq} = |p - q|.$$

Допустимі розклади можна оцінити за сумарними витратами на повторні відвідування потягів, розділені переміщення контейнерів та безпосередні перевантаження контейнерів між потягами в межах одного слота. Тому критерій ефективності розкладу наведемо у вигляді

$$\Psi'(X) = \alpha_1 \sum_{i \in I} C^r y_i + \alpha_2 \sum_{t=1}^T \sum_{t'=t+1}^T \sum_{p=1}^G \sum_{q=1}^G z_{tt'} A_{\kappa_p^t \kappa_q^{t'}} C_{pq}^* +$$

$$+ \alpha_3 \sum_{t=1}^T \sum_{p=1}^G \sum_{q=1}^G A_{\kappa_p^t \kappa_q^t} C_{pq},$$
(4.5)

де  $I = \{1, 2, \dots, N\}$  – множина потягів;  $A = [A_{ij}]$ ,  $i, j = 1, 2, \dots, N$  – кількість контейнерів, які потяг  $i$  перевозить для потягу  $j$ ;  $\alpha_1, \alpha_2, \alpha_3$  – вагові коефіцієнти витрат на повторні відвідування потягів, розділені переміщення контейнерів та безпосередні перевантаження контейнерів між потягами в межах одного слота.

Математична модель задачі має вигляд

$$\min_{X \in \mathbb{W}} \Psi'(X),$$

$$\mathbb{W} = \{X \in \mathbb{R}^{G \cdot T}\},$$

$$X = (K^1, K^2, \dots, K^T) \in \mathbb{R}^{G \cdot T},$$

$$K^t = (\kappa_1^t, \kappa_2^t, \dots, \kappa_G^t) \in E_N^G \subset R^G, \kappa_g^t \in I, g = 1, 2, \dots, G,$$

$$t = 1, 2, \dots, T,$$
(4.6)

де допустимі розв'язки  $X$  задовольняють обмеження (4.1) та (4.2).

Метод розв'язання наведеної задачі та обчислювальні експерименти описано в праці [42].

#### 4.3. Задача розміщення контейнерів за умов експлуатації залізничного термінала

У підрозділі досліджено загальну проблему планування роботи залізничного термінала на рівні III. Запропоновано вибрати оптимальне за

вартістю розташування контейнерів на залізничних платформах та в зоні зберігання під час оброблення потягів на залізничному терміналі.

Розміщення контейнерів на залізничному терміналі передбачає перенесення контейнерів безпосередньо з потягу на потяг, від потягу до зони зберігання або від зони зберігання до потягу з вибором місця для кожного контейнера. Вартість операцій з перероблення контейнерів визначається відстанню на переміщення контейнерів. Оптимізація оброблення контейнерів на рівні III полягає у зменшенні витрат на переміщення контейнерів на основі вибору їхніх позицій на платформах та в зоні зберігання.

Отже, проблема розміщення контейнерів у разі оброблення потягів на залізничному терміналі полягає у визначенні оптимального розташування контейнерів у цільових потягах та в зоні зберігання на основі вирішення проблеми рівня I (тобто задачі TYSP). Використовуючи сформовані часові слоти обслуговування, а також відомі положення потягів на коліях на станції (розв'язок задачі знаходження оптимального розкладу прибуття вантажних потягів та перевантаження контейнерів на залізничному терміналі з призначенням потягів на колії, див. п. 4.2), необхідно визначити оптимальні положення контейнерів на цільових потягах та в зоні зберігання.

У кожному сервісному слоті певні контейнери з потягів-джерел та визначені контейнери із зони зберігання повинні бути переміщені та розміщені на платформах у відповідних цільових потягах. Приймальні платформи в цільових потягах слід вибирати з вільних платформ та платформ, на яких контейнери призначені для переміщення до зони зберігання. Зазначені контейнери з потягів-джерел слід перемістити у вільні місця в зоні зберігання.

Проблема полягає в побудові розкладу оброблення потягів (віднесення потягів до слотів), визначенні місць розташування потягів на коліях залізничного терміналу (призначення потягів на колії) та складанні розкладу розміщення контейнерів з метою мінімізації витрат на обслуговування потягів.

Запропоновано стратегію розв'язання, яка містить два етапи: на першому етапі розв'язують задачу планування руху вантажних потягів на

залізничному терміналі із формуванням сервісних слотів та призначенням потягів на колії. Математична модель і метод розв'язання задачі запропоновані в [42] і викладені в п. 4.2. На другому етапі для кожного сервісного слота, сформованого на першому етапі, визначають оптимальні позиції для контейнерів у цільових потягах та в зоні зберігання згідно з умовою мінімізації загальних витрат на оброблення контейнерів.

Сформулюємо задачу, яка має бути розв'язана на другому етапі запропонованої стратегії для кожного сервісного слота.

*Задача розміщення контейнерів за умов експлуатації залізничного терміналу (Containers Placement In Rail Terminal Operations Problem, CPRTOP).* Для заданого сервісного слота вантажних потягів визначити оптимальні позиції контейнерів у цільових потягах та в зоні зберігання на залізничному терміналі, мінімізуючи загальні витрати на переміщення контейнерів.

Побудуємо математичну модель задачі CPRTOP. Уведемо такі параметри і змінні для формалізації задачі:

$G$  – кількість колій та кількість вантажних потягів у сервісному слоті;

$J_n = \{1, 2, \dots, n\}$  – індексна множина;

$N = [n_{ij}]$ ,  $i, j \in J_G$  – кількість контейнерів у потязі-джерелі  $i$ , призначених для цільового потягу  $j$ ;

$l_j$ ,  $j \in J_G$  – кількість контейнерів у зоні зберігання, призначених для цільового потягу  $j$ ;

$k_i$ ,  $i \in J_G$  – кількість контейнерів у потязі-джерелі  $i$ , призначених для переміщення в зону зберігання;

$m_j$ ,  $j \in J_G$  – кількість вільних платформ у цільовому потязі  $j$ ;

$d$  – кількість вільних позицій для контейнерів у зоні зберігання.

Нехай  $A$  – множина контейнерів, пов'язаних з операціями залізничного терміналу, на поточному слоті обслуговування,  $A = \{a_1, a_2, \dots, a_D\}$ . Кожний контейнер  $a_i \in A$  опишемо кортежем 4 параметрів,  $a_i = (a_x^i, a_y^i, a_f^i, a_t^i)$ . Тут  $(a_x^i, a_y^i)$  – координати контейнера  $a_i$  у потязі-джерелі  $a_f^i \in J_G$  або в зоні зберігання ( $a_f^i = 0$ ), призначеного для розміщення в цільовому потязі  $a_t^i \in J_G$  або в зоні

зберігання ( $a_t^i=0$ ). Зобразимо вільні платформи в цільових потягах і вільні місця в зоні зберігання контейнерів тими самими засобами, що і контейнери.

Упорядкуємо всі контейнери з множини  $A$ :  $A_1, A_2, \dots, A_G, A_{G+1}$ . Тут множина  $A_i$  складається з контейнерів потягу  $i$  у поточному слоті,  $i \in J_G$ . Елементи множини  $A_{G+1}$  є контейнерами із зони зберігання, призначеними для потягів  $1, 2, \dots, G$ . Впорядкуємо всі контейнери з множини  $A_i$  в такий спосіб:

$$A_i = \{a_1^i, a_2^i, \dots, a_{n_1}^i \mid a_{n_1+1}^i, \dots, a_{n_1+n_2}^i \mid \dots \mid a_{n_1+\dots+n_{G-1}}^i, \dots, a_{n_1+\dots+n_G}^i \mid \\ \mid a_{N_1+1}^i, \dots, a_{N_1+k_i}^i \mid a_{N_1+k_i+1}^i, \dots, a_{N_1+k_i+m_i}^i \},$$

де  $a_1^i, a_2^i, \dots, a_{n_1}^i$  – контейнери потягу  $i$ , призначені для потягу ( $i$  колії)  $j \in J_G$ ,  $1, \dots, a_{n_1+\dots+n_{G-1}}^i, \dots, a_{n_1+\dots+n_G}^i$  – контейнери для потягу  $G$ ,  $a_{N_1+1}^i, \dots, a_{N_1+k_i}^i$  – контейнери потягу  $i$ , призначені для переміщення в зону зберігання,  $a_{N_1+k_i+1}^i, \dots, a_{N_1+k_i+m_i}^i$  – вільні платформи в потязі  $i$ .

Визначимо змінні моделі. Нехай  $U = \{u \in R^\sigma\}$ ,  $u = (u_1, u_2, \dots, u_G)$ , де  $u_i = (X_i, Y_i, Z_i)$ ,  $X_i = (X_1^i, X_2^i, \dots, X_{N_i}^i)$ ,  $X_i^t = (x_1^{it}, x_2^{it}, \dots, x_{n_{it}}^{it}) \in A_{m_t}^{n_{it}}$ ,  $N_i = \sum_{j=1}^G n_{ij}$ ,

$A_{m_t}^{n_{it}}$  – множина розміщень  $m_t$  елементів, що відповідають вільним платформам у цільовому потязі  $t$  по  $n_{ij}$  контейнерів, що дорівнює кількості контейнерів у потязі-джерелі  $i$ , призначеному для цільового потягу  $j$ ,  $x_j^{it} = (x_x^{itj}, x_y^{itj})$ ;  $Z_i = (z_1^i, z_2^i, \dots, z_{k_i}^i) \in A_{d_i}^{k_i}$ , де  $z_j^i = (z_x^{ij}, z_y^{ij})$  – позиції в зоні

зберігання для  $k_i$  контейнерів з потягу-джерела  $i$ ,  $d_i = d - \sum_{j=1}^{i-1} k_j$ ;  $Y_i$  – вільні

платформи у потязі  $i$  для контейнерів з усіх потягів-джерел, для яких потяг  $i$  є цільовим потягом  $y_j^i = (y_x^{ij}, y_y^{ij})$ :

$$Y_i = (y_1^i, y_2^i, \dots, y_{\bar{N}_i}^i, y_{\bar{N}_i+1}^i, \dots, y_{\bar{N}_i+l_i}^i) \in A_{m_i-n_{it}}^{\bar{N}_i+l_i}, \quad \bar{N}_i = \sum_{j=1}^G n_{ji},$$

$$\sigma = \sum_{i=1}^G (\bar{N}_i + l_i + k_i + \sum_{t=1}^{N_i} n_{it}).$$

Вважаємо, що витрати на експлуатацію залізничного термінала визначаються відстанями, пройденими козовим краном. Щоб зменшити загальну вартість переміщення контейнерів, потрібно мінімізувати загальну відстань, яку подолав козовий кран за всі операції залізничного термінала. Враховуючи специфіку руху козового крана (вздовж двох ортогональних осей), використовуємо мангеттенську відстань (метрику) для розрахунку довжини шляху, пройденого краном. На підставі цього загальна вартість контейнерних операцій з потягом  $i$  дорівнює

$$F_i(u) = \sum_{t=1}^{N_i} \sum_{s=1}^{n_{it}} \rho(a_{n_{i1}+\dots+n_{it-1}+s}^i, x_s^{it}) + \sum_{s=1}^{k_i} \rho(a_{N_1+s}^i, z_s^i) +$$

$$+ \sum_{t=1}^{G+1} \sum_{s=1}^{n_{iti}} \rho(y_s^i, a_{n_{1i}+n_{12}+\dots+n_{i-1i}+s}^i), \quad (4.7)$$

де  $\rho(a_i, x_i) = |a_x^i - x_x^i| + |a_y^i - x_y^i|$ .

Загальна вартість операцій з усіма потягами для поточного слота обслуговування становить:

$$F(u) = \sum_{i=1}^G F_i(u).$$

Згідно з уведеними позначеннями та припущеннями запишемо математичну модель задачі CPRTOP у такому вигляді:

$$F(u) = \sum_{i=1}^G F_i(u) \rightarrow \min, \quad (4.8)$$

$$u \in U \subset R^\sigma. \quad (4.9)$$

Ця задача комбінаторної оптимізації є NP-складною і може бути розв'язана як за допомогою точних, так і евристичних методів. Деякі підходи до розв'язання задач комбінаторної оптимізації на основі генерації комбінаторних конфігурацій зі спеціальними властивостями наведено в праці [156].

Побудова математичної моделі задачі CPRTOP є першим кроком у дослідженні процесу оптимізації розміщення контейнерів на залізничному терміналі. Наступним етапом дослідження має стати аналіз математичної моделі одним із методів комбінаторної оптимізації. Оскільки результати, викладені в цьому підрозділі, є продовженням досліджень, наведених у п. 4.2 для задачі на другому рівні, для аналізу математичної моделі може бути використана модифікація евристичного методу, запропонована в [42]. Програмне забезпечення, розроблене на базі результатів обох етапів дослідження, може бути використане для планування руху потягів на сучасному залізничному терміналі.

Приклад 4.1. Нехай сервісний слот складається з двох потягів на двох коліях і  $n_{12}=0, n_{21}=2; l_1=2, l_2=0; k_1=0, k_2=2; m_1=4, m_2=0; d=3; A=\{a_1, a_2, a_3, a_4, a_5, a_6\}$ . Координати контейнерів наведено в таб. 4.1. Визначимо вартість двох різних варіантів переміщення контейнерів у слоті обслуговування.

*Перший варіант.* Вартість переміщення контейнера  $a_1$  до платформи  $x_3$  і контейнера  $a_2$  до платформи  $x_4$  дорівнює  $C_1=\rho(a_1, x_3)+\rho(a_2, x_4)=|1-4|+|2-1|+|3-5|+|2-1|=7$ . Вартість переміщення контейнера  $a_4$  до платформи  $y_1$  і контейнера  $a_3$  до платформи  $y_2$  дорівнює  $C_2=\rho(a_4, y_1)+\rho(a_3, y_2)=|1-2|+|0-1|+|2-1|+|0-1|=4$ . Вартість переміщення контейнера  $a_5$  до платформи  $z_1$  і контейнера  $a_6$  до платформи  $z_2$  дорівнює  $C_3=\rho(a_5, z_1)+\rho(a_6, z_2)=|2-3|+|2-0|+|4-1|+|2-0|=5$ .

Загальна вартість операцій з усіма потягами для поточного слота обслуговування становить  $F(u)=C_1+C_2+C_3=7+5+4=16$ .

Таблиця 4.1. Координати контейнерів

Контейнери	Вільні платформи
$a_1=(1,2,2,1)$ $a_2=(3,2,2,1)$	$x_1=(1,1)$ $x_2=(2,1)$ $x_3=(4,1)$ $x_4=(5,1)$
$a_3=(1,0,0,1)$ $a_4=(2,0,0,1)$	$y_1=(1,1)$ $y_2=(2,1)$
$a_5=(2,2,2,0)$ $a_6=(4,2,2,0)$	$z_1=(3,0)$ $z_2=(4,0)$ $z_3=(5,0)$

*Другий варіант.* Обчислимо значення цільової функції, якщо розміщення контейнерів  $a_3$  і  $a_4$  змінюються. Контейнер  $a_3$  переміщується на платформу  $y_1$  і контейнер  $a_4$  переміщується на платформу  $y_2$ , тоді  $C_3=\rho(a_3, y_1)+\rho(a_4, y_2)=|1-1|+|0-1|+|2-2|+|0-1|=2$ .

Загальна вартість операцій з усіма потягами для поточного слота обслуговування дорівнює  $F(u)=C_1+ C_2+ C_3=7+5+2=14$ .

Отже, зміна координат розміщення контейнерів зменшує загальну вартість переміщення контейнерів на залізничному терміналі.

#### 4.4. Розміщення контейнерів різного типу на залізничному контейнерному терміналі за допомогою Double-Stack завантаження

Під час оброблення контейнерів у залізничному контейнерному терміналі використовуються різні типи контейнерів.

За стандартом ISO визначаються уніфіковані розміри контейнерів для зручного перевезення різними видами транспорту. Зазвичай, контейнери характеризуються такими основними параметрами: розмір [65, 232, 233] (довжина, ширина, висота), вага та тип. Найпоширенішими стандартними розмірами контейнерів є:

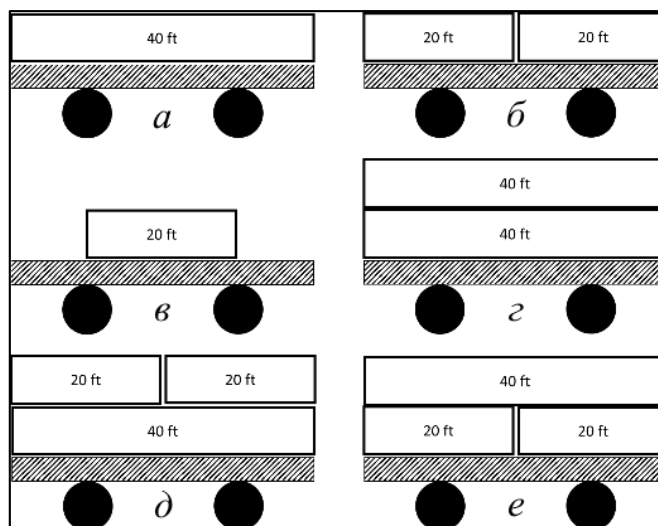
- 20-футовий стандартний контейнер (20' - 8' - 8'6");

- 40-футовий стандартний контейнер (40' - 8' - 8'6");
- 40-футовий високий контейнер (40' - 8' - 9'6");
- 45-футовий високий контейнер (45' - 8' - 9'6").

Для перевезення таких контейнерів на залізниці використовуються спеціальні платформи з кріпленнями (well car, double-stack car). Вони бувають різних розмірів, що відповідають розмірам стандартних контейнерів. Найпоширеніші довжини платформ – 40 футів, 48 футів, 53 фути [234].

Контейнери на залізничних платформах можуть бути розміщені один на одному. Таке розміщення має назву double-stack. При цьому використовуються поняття “top slot” – місце для контейнерів, розміщених зверху, та “bottom slot” – місце для контейнерів, розміщених знизу [234].

У контейнерних перевезеннях використовують такі основні варіанти розміщення контейнерів на залізничних платформах та в зоні зберігання контейнерного терміналу: розміщення одного контейнера на одній залізничній платформі відповідного розміру (рис. 4.2, *a*); розміщення двох 20-футових контейнерів поряд на одній залізничній платформі (рис. 4.2, *б*); розміщення одного контейнера на одній залізничній платформі більшого розміру (рис. 4.2, *в*); розміщення двох контейнерів, однакових за розміром, один на одному в двох слотах (double-stack) (рис. 4.2, *г*); розміщення трьох контейнерів у двох слотах у разі розташування в одному слоті двох, менших за розміром, контейнерів (double-stack) (рис. 4.2, *д*, *е*).



**Рис. 4.2.** Основні варіанти розміщення контейнерів на залізничних платформах

Задача, що досліджується, базується на загальній проблемі планування роботи перевантажувальної станції (див. п. 4.1) та є узагальненням задачі розміщення контейнерів за умов експлуатації залізничного терміналу (див. п. 4.3) на випадок застосування контейнерів різних стандартних розмірів та можливості завантаження double-stack.

Четвертий рівень розв'язання загальної задачі планування роботи перевантажувальної станції – призначення контейнерів, які необхідно перевантажити в рамках одного сервісного слота, на порожні платформи в потягах-приймачах та в зоні зберігання. Задачу контейнерних переміщень на залізничній перевантажувальній станції (CPRTOP) для однакових за розміром контейнерів та платформ розв'язано в п. 4.3. Результатом розв'язання задачі є визначені місця (вільні платформи) на потягах-приймачах або в зоні зберігання для кожного контейнера з потягів-джерел, які обслуговуються в одному сервісному слоті.

Розглянемо проблему оптимізації процесу переміщення різних типів контейнерів під час оброблення потягів на перевантажувальній станції з можливістю використання double-stack завантаження. Для сформованого сервісного слота з потягами, які призначені на певні колії, та контейнерами,

для яких визначені потяги-приймачі, потрібно мінімізувати вартість перевантаження контейнерів у межах сервісного слота. При цьому слід враховувати умови, що контейнери, які обслуговуються, мають різні розміри та можуть бути розміщені в різних слотах (double-stack) на однакових платформах стандартної довжини.

Однією з особливостей роботи залізничного контейнерного терміналу є робота козлового крана, який безпосередньо перевантажує контейнери з потягу на потяг (або у зону зберігання). Зазначимо, що порталний кран може рухатися лише вздовж двох ортогональних осей: порталний кран переміщується вздовж колій; механізм «захоплення» контейнера переміщується перпендикулярно коліям.

З огляду на це сформулюємо задачу дослідження.

*Задача розміщення контейнерів різних типів з використанням двоярусного завантаження (Different types containers double-stack placement problem, DTDS CPP).* Для заданого сервісного слота вантажних потягів із заздалегідь визначеною множиною контейнерів, що мають різні характеристики, та з відомою можливістю двоярусного завантаження для кожного потягу необхідно визначити оптимальні позиції контейнерів на стандартних платформах однакової довжини в цільових потягах та в зоні зберігання контейнерного терміналу, мінімізуючи загальні витрати на обслуговування контейнерів. Загальна вартість обслуговування включає вартість переміщення контейнера та вартість використання слота (платформи).

Під час розв'язання задачі розміщення різних типів контейнерів на залізничному контейнерному терміналі з використанням double-stack враховуємо, що обслуговуються контейнери двох видів (20 ft standard and 40 ft standard), 40-футові платформи для перевезення контейнерів та 40-футові порожні місця в зоні зберігання. Для кожного потягу визначено можливість застосування double-stack завантаження (можливий чи неможливий double-stack).

**Математична модель.** Для побудови математичної моделі сформульованої задачі DTDS CPP уведемо такі позначення:

- $I_n = \{1, 2, \dots, n\}$  – індексна множина;
- $S$  – кількість паралельних колій на залізничному терміналі;
- $V = \{v_1, \dots, v_M\}$  – множина сервісних слотів, кожний слот містить  $S$  потягів, які обслуговуються одночасно;
- $D_v = D_{v_{20}} + D_{v_{40}}$  – загальна кількість 20-футових та 40-футових контейнерів на потягах, які потрібно перевантажити в сервісному слоті  $v$ ;
- $D_{v_{20}}$  – кількість 20-футових контейнерів, які потрібно перевантажити в слоті  $v$ ;
- $D_{v_{40}}$  – кількість 40-футових контейнерів, які потрібно перевантажити в слоті  $v$ ;
- $m_j^{20}, m_j^{40}, j \in I_s$ , де  $m_j^{20}$  кількість 20-футових та  $m_j^{40}$  40-футових, які знаходяться в зоні зберігання та призначені для потягу-приймача  $j$  в поточному сервісному слоті;
- $N^{20} = [n_{ij}^{20}]$  та  $N^{40} = [n_{ij}^{40}]$ ,  $i, j \in I_s$ , де  $N^{20}$  – кількість 20-футових контейнерів та  $N^{40}$  – кількість 40-футових контейнерів на потязі-джерелі  $i$ , призначених потягу-приймачу  $j$  в поточному сервісному слоті;
- $p_i^{20}, p_i^{40}, i \in I_s$  – де  $p_i^{20}$  – кількість 20-футових та  $p_i^{40}$  – кількість 40-футових контейнерів на потязі-джерелі  $i$  в поточному сервісному слоті, які призначені для зони зберігання;
- $r_j, j \in I_s$  – кількість вільних платформ на потязі  $j$ ;
- $q$  – кількість вільних місць в зоні зберігання.

Оскільки задача розміщення різних типів контейнерів передбачає, зокрема, можливість double-stack розміщення, введемо булеву змінну  $l_i$ , яка визначає можливість розміщення верхнього слота (top slot) на потязі  $i$ :

$$l_i = \begin{cases} 0, & \text{верхній слот не може бути розміщений,} \\ 1, & \text{верхній слот може бути розміщений.} \end{cases}$$

Уведемо множину  $C = \{c_1, c_2, \dots, c_{D_v}\}$  контейнерів, які обслуговуються на перевантажувальній станції в одному сервісному слоті. Кожний контейнер  $c_i \in C$  можна описати такими параметрами: координати контейнера  $c_i - c_x^i, c_y^i$ , які характеризують його положення на перевантажувальній станції; номер потягу-джерела, на якому знаходиться контейнер –  $c_f^i \in I_s, c_f^i = 0$ , якщо контейнер знаходиться в зоні зберігання; номер цільового потягу, на який контейнер потрібно перевантажити –  $c_g^i \in I_s, c_g^i = 0$ , якщо контейнер перевантажується в зону зберігання; розмір (тип) контейнера –  $c_u^i$ . Тобто будь-який контейнер  $c_i \in C$  описують кортежем з п'яти параметрів  $c_i = (c_x^i, c_y^i, c_f^i, c_g^i, c_u^i)$ .

Множину контейнерів  $C$  розіб'ємо на підмножини  $C_1, C_2, \dots, C_S, C_{S+1}$  відповідно до потягів-джерел, на яких вони знаходяться. Тут  $C_i, i \in I_s$  – множина контейнерів на потязі  $i$ ,  $C_{S+1}$  – множина контейнерів із зони зберігання, призначених потягам  $1, 2, \dots, S$ . У кожній множині  $C_i$  виділено підмножини  $C_i'$  та  $C_i''$ ,  $C_i = C_i' \cup C_i''$ ,  $C_i'$  – множина 20-футових контейнерів, а  $C_i''$  – множина 40-футових контейнерів на потязі  $i$ . Кожну множину  $C_i$  можна описати так:

$$C_i' = \{c_1^i, c_2^i, \dots, c_{n_{11}^{20}}^i \mid c_{n_{11}^{20}+1}^i, \dots, c_{n_{11}^{20}+n_{12}^{20}}^i \mid c_{n_{11}^{20}+\dots+n_{1S-1}^{20}}^i, \dots, c_{n_{11}^{20}+\dots+n_{1S}^{20}}^i \mid c_{N_1^{20}+1}^i, \dots, c_{N_1^{20}+p_i^{20}}^i \mid c_{N_1^{20}+p_i^{20}+1}^i, \dots, c_{N_1^{20}+p_i^{20}+r_i}^i\},$$

$$C_i'' = \{c_1^i, c_2^i, \dots, c_{n_{11}^{40}}^i \mid c_{n_{11}^{40}+1}^i, \dots, c_{n_{11}^{40}+n_{12}^{40}}^i \mid c_{n_{11}^{40}+\dots+n_{1S-1}^{40}}^i, \dots, c_{n_{11}^{40}+\dots+n_{1S}^{40}}^i \mid c_{N_1^{40}+1}^i, \dots, c_{N_1^{40}+p_i^{40}}^i \mid c_{N_1^{40}+p_i^{40}+1}^i, \dots, c_{N_1^{40}+p_i^{40}+r_i}^i\},$$

де  $c_1^i, c_2^i, \dots, c_{n_{11}^{20}}^i$  та  $c_1^i, c_2^i, \dots, c_{n_{11}^{40}}^i$  – 20- та 40-футові контейнери, розташовані на потязі  $i$  та призначені для потягу  $j, j \in I_s$ ;  $c_{n_{11}^{20}+\dots+n_{1S-1}^{20}}^i, \dots, c_{n_{11}^{20}+\dots+n_{1S}^{20}}^i$  та  $c_{n_{11}^{40}+\dots+n_{1S-1}^{40}}^i, \dots, c_{n_{11}^{40}+\dots+n_{1S}^{40}}^i$  – 20- та 40-футові контейнери для потягу  $S$ ;  $c_{N_1^{20}+1}^i, \dots, c_{N_1^{20}+p_i^{20}}^i$  та  $c_{N_1^{40}+1}^i, \dots, c_{N_1^{40}+p_i^{40}}^i$  – 20- та 40-футові контейнери з потягу  $i$ ,

які потрібно перемістити у зону зберігання;  $c_{N_i^{20}+p_i^{20}+1}^i, \dots, c_{N_i^{20}+p_i^{20}+r_i}^i$  та  $c_{N_i^{40}+p_i^{40}+l}^i, \dots, c_{N_i^{40}+p_i^{40}+r_i}^i$  – вільні платформи на потязі  $i$ .

Для опису можливих варіантів призначення контейнерів на платформах та в зоні зберігання в математичній моделі використано комбінаторну множину розміщень  $A_n^m$  – упорядкованих  $m$ -елементних наборів з множини  $n$  елементів; для формування варіантів неупорядкованих підмножин контейнерів – комбінаторну множину комбінацій  $C_n^m$  – неупорядкованих  $m$ -елементних наборів з множини  $n$  елементів [110].

Для математичної моделі задачі визначимо такі змінні:

- $X = \{x \in R^\sigma / x = (x_1, x_2, \dots, x_S)\}$ , де  $x_i = (B_i, T_i, G_i, H_i)$ ;
- $\sigma = \sum_{i=1}^S \bar{N}_i + m_i + p_i + \sum_{v=1}^{N_i} n_{iv}$ ,  $\bar{N}_i = \sum_{j=1}^S n_{ij}$ ;
- $B_i = (B_1^i, B_2^i, \dots, B_{N_i}^i)$ ,  $B_i^v = (b_1^{iv}, b_2^{iv}, \dots, b_{n_{iv}}^{iv}) \in A_{r_v}^{n_{iv}}$ ,  $N_i = \sum_{j=1}^S n_{ij}$ , де  $A_{r_v}^{n_{iv}}$  – множина розміщень  $r_v$  порожніх платформ у потязі-приймачі  $j$  по  $n_{iv}$  контейнерів з потягу-джерела  $i$ , які призначені потягу-приймачу  $j$ ,  $b_j^{iv} = (b_x^{ivj}, b_y^{ivj})$ ;
- $T_i = (t_1^i, t_2^i, \dots, t_{p_i}^i) \in A_{q_i}^{p_i}$ , де  $t_j^i = (t_x^{ij}, t_y^{ij})$  – позиції в зоні зберігання для  $p_i$  контейнерів з потягу-джерела  $i$ ,  $q_i = q - \sum_{j=1}^{i-1} p_j$ ;
- $G_i$  – порожні платформи на потязі  $i$  для контейнерів, які необхідно перевантажити з інших потягів-джерел на потяг-приймач  $i$ ,  $g_j^i = (g_x^{ij}, g_y^{ij})$ ,  $G_i = (g_1^i, g_2^i, \dots, g_{\bar{N}_i}^i, g_{\bar{N}_i+1}^i, \dots, g_{\bar{N}_i+m_i}^i) \in A_{r_i-n_{ii}}^{\bar{N}_i+m_i}$ ;
- $H_i = \{H_1^i, H_2^i, \dots, H_z^i\} = C_{E(v)}^2$ ,  $H_i^v = (h_1^{iv}, h_2^{iv}) \in C_{E(v)}^2$ , де  $C_{E(v)}^2$  – множина комбінацій пар 20-футових контейнерів, призначених для потягу  $j$ , з потягу  $i$ ,  $E(v) = D_{v20}^{n_i}$ ,  $z = |C_{E(v)}^2|$ .

Оцінка вартості перевантаження контейнерів під час обслуговування потягів на залізничному терміналі визначається відстанню, яку необхідно подолати порталюному крану, та завантаженістю платформ у рамках сервісного слота. Для розв'язання задачі необхідно мінімізувати загальну вартість обслуговування потягів у сервісному слоті.

Враховуючи особливості переміщення крана, можна застосувати мангеттенську метрику для визначення відстані, яку він проходить:

- $z_1(c_i, b_j) = |c_x^i - b_x^j| + |c_y^i - b_y^j|$  – відстань, яку проходить порталний кран у разі переміщення одного 40-футового контейнера  $c_i$  з потягу-джерела на платформу  $b_j$  цільового потягу;

- $z_2(c_i, c_k, b_j) = z_1(c_i, b_j) + z_1(c_k, b_j)$  – загальна відстань, яку проходить порталний кран у разі переміщення двох 20-футових контейнерів  $c_i$  та  $c_k$  з потягу-джерела на платформу  $b_j$  цільового потягу;

- $o$  – вартість переміщення контейнера на одиницю відстані;
- $\bar{u}$  – вартість розміщення контейнера на платформі (використання платформи).

Завантаженість (використання) платформи визначається такими умовами:

- платформа вважається порожньою, якщо: 1) на ній немає жодного контейнера (порожні bottom slot та top slot); 2) якщо на платформі розташований один 40-футовий контейнер чи два 20-футових контейнера (порожній top slot). Умова 2) може бути застосована, якщо для потягу дозволено double-stack завантаження. У випадку перевезення порожньої платформи застосовується штраф  $\bar{u}$ ;

- платформа вважається зайнятою, якщо: 1) зайняті bottom slot та top slot; 2) зайнятий bottom slot – для потягів без можливого double-stack завантаження;

- платформа вважається частково завантаженою, якщо на ній розташований один 20-футовий контейнер. У випадку перевезення частково завантаженої платформи застосовується штраф  $\bar{u}''$ .

Розглянемо випадок, коли пари 20-футових контейнерів сформовані заздалегідь, базуючись на евристичному підході. У такому випадку  $d_i = \lceil (p_i^{20} + 1) / 2 \rceil$  – кількість пар 20-футових контейнерів, які призначені для

зони зберігання;  $e_{ij} = \lfloor (n_{ij}^{20} + 1) / 2 \rfloor$  – кількість пар 20-футових контейнерів на потязі-джерелі  $i$ , які призначені для потягу-приймача  $j$ ;  $s_{ji} = \lfloor (n_{ji}^{20} + 1) / 2 \rfloor$  – кількість пар 20-футових контейнерів, які потрібно перемістити на цільовий потяг  $i$ .  $(c_i^o, c_i^p)$  – пари контейнерів, які призначені для порожньої платформи на цільовому потязі  $i$ ,  $i \in I_s$  або зони зберігання, коли  $i = S + 1$ .

Тоді відстань, яку проходить козловий кран під час переміщення контейнерів у разі обслуговування потягу  $i$  в слоті  $v$ , визначають за формулою:

$$\begin{aligned}
 f_i(x) = & \sum_{k=1}^{p_i^{40}} z_1(c_k^i, t_k^i) + \sum_{a=1}^{d_i} z_2(c_a^o, c_a^p, t_a^i) + \\
 & + \sum_{j=1}^S \sum_{k=1}^{n_{ij}^{40}} z_1(c_{n_{i1}^{40} + \dots + n_{ij-1}^{40} + k}^i, b_k^{iv}) + \sum_{j=1}^S \sum_{t=1}^{e_{ij}} z_2(c_t^o, c_t^p, b_t^{ij}) + \\
 & + \sum_{j=1}^{S+1} \sum_{k=1}^{n_{ji}^{40}} z_1(c_{n_{j1}^{40} + n_{j2}^{40} + \dots + n_{j-1}^{40} + k}^i, g_k^i) + \sum_{j=1}^{S+1} \sum_{t=1}^{n_{ji}^{20}} z_2(c_t^o, c_t^p, g_t^i),
 \end{aligned}$$

Загальну відстань, яку проходить козловий кран під час переміщення контейнерів у разі обслуговування всіх потягів у сервісному слоті  $v$  визначають так:

$$F(x) = \sum_{i=1}^S f_i(x).$$

Математична модель задачі DTDS CPP має вигляд комбінаторної задачі оптимізації:

$$\begin{aligned}
 F(x) = \sum_{i=1}^S f_i(x) \rightarrow \min, \\
 x \in X \subset R^\sigma.
 \end{aligned}$$

**Стратегія розв'язання задачі.** Стратегію евристичного розв'язання задачі DTDS CPP наведемо у вигляді послідовності етапів. Для кожного потягу-джерела:

1. Визначення кількості та позицій контейнерів, які потрібно перевантажити з даного потягу в межах сервісного слота;

2. Виділення підмножини 20-футових контейнерів та формування оптимальних за вартістю пар для розміщення на одній вільній платформі відповідного потягу-приймача;

3. Оптимальне розміщення сформованих пар 20-футових контейнерів, що відповідає мінімальній сумарній вартості;

4. Оптимальне розміщення 40-футових контейнерів, що відповідає мінімальній сумарній вартості.

Виконуючи кроки 1–4, зміну стану завантаженості платформи у разі переходу до наступного потягу-джерела необхідно враховувати алгоритмічно. Здійснення кроків 3 та 4 можливо на базі підходу, запропонованого в [57].

4.5. Оптимізація порядку переміщення контейнерів на залізничних терміналах

П'ятим рівнем вирішення загальної проблеми планування роботи залізничного терміналу (тобто задачі TYSP) є визначення послідовності переміщення контейнерів краном (див. п. 4.1).

Розглядаючи задачу оптимізації порядку переміщення контейнерів на залізничних терміналах, приймемо такі гіпотези:

1. Вартість переміщення контейнерів від потягу-джерела до цільового потягу залежить від відстані між потягами.

2. Якщо потяг-джерело та цільовий потяг знаходяться в одному сервісному слоті, вони повинні розташовуватися на найближчих коліях.

3. Якщо потяги належать до різних слотів обслуговування, необхідно мінімізувати хід козлового крана.

4. Вартість переміщення контейнерів оцінюється відстанню, яку проїжджає козловий кран.

Метою розв'язання задачі є визначення оптимального порядку переміщення контейнерів між потягами, або між потягом та зоною зберігання у сформованих сервісних слотах.

**Постановка задачі.** Розглядаємо множину потягів з визначеною кількістю та координатами порожніх платформ та контейнерів на цих потягах. Усі потяги, що прибувають до залізничного терміналу, мають бути призначені до сервісних слотів. Кількість потягів, які можна обслуговувати одночасно (в одному слоті), не повинна перевищувати кількості паралельних колій на залізничному контейнерному терміналі. Оптимальний слот обслуговування побудований як розв'язок комбінаторної задачі оптимізації. Під час обслуговування потягів в одному сервісному слоті контейнери повинні перевантажуватися від потягу до потягу, від потягу до зони зберігання або від зони зберігання до потягу. Враховуючи координати розташування контейнерів для переміщення та координати вільних платформ у цільових потягах та в зоні зберігання, проблема оптимального розміщення контейнерів вирішується як задача комбінаторної оптимізації.

На основі результатів розв'язання задачі на рівнях 1–3 (див. п. 4.2, 4.3) припускаємо, що доступні та відомі такі дані: потяги, призначені в один слот обслуговування; призначення кожного потягу на залізничну колію; призначення кожного контейнера до певної вільної платформи в цьому слоті обслуговування. Тому задачу оптимізації порядку переміщення контейнерів на залізничному терміналі можна сформулювати так.

**Задача.** *Порядок переміщення контейнерів на залізничному терміналі (Relocation Order of Containers at a Transshipment Yard, ROCTY).* Нехай визначено сервісний слот із заданим положенням потягів на коліях залізничного терміналу, відомі координати контейнерів та вільних платформ на потягах та в зоні зберігання; для кожного контейнера відомо 2-D координати платформи, на яку цей контейнер необхідно перемістити.

Визначити порядок переміщення контейнерів порталним краном, мінімізуючи витрати на обслуговування всього сервісного слота.

*Математична модель задачі оптимізації порядку розміщення контейнерів на залізничних терміналах.* Нехай  $V = \{v_1, v_2, \dots, v_n\}$ , де  $v_i = (v_x^i, v_y^i)$  – множина координат геометричних центрів контейнерів, які потрібно перемістити, на потязі-джерелі або в зоні зберігання та координати вільних платформ на цільових потягах або в зоні зберігання,  $n$  – це кількість контейнерів та порожніх платформ. Сформуємо множину  $D = \{d_1, d_2, \dots, d_m\}$ , де  $d_i = (d_i^f, d_i^t)$  – впорядковані пари елементів з множини  $V$ , які описують переміщення кожного контейнера з координатами  $d_i^f$  на порожню платформу з координатами  $d_i^t$ ,  $m$  – це кількість контейнерів, які необхідно перемістити.

Задамо змішаний граф  $G = (V, E)$ , де  $V = \{v_1, v_2, \dots, v_n\}$  – це множина вершин,  $E = D \cup C$  – об'єднання множини дуг  $D$  (відповідають переміщенню контейнерів з місця на місце) та ребер  $C$  (які відповідають переміщенню порталного крана без контейнера). Множина ребер  $C$  побудована так, що граф  $G_1 = (V, C)$  є повним неорієнтованим. Для графа  $G$  обчислюємо матрицю довжин ребер  $\|c_{ij}\|$ , яка базується на відстані між вершинами графа та розраховується за мангеттенською метрикою:  $c_{ij} = |v_x^i - v_x^j| + |v_y^i - v_y^j|$ ,  $i, j = \{1, 2, \dots, n\}$ .

Перш за все, потрібно обрати початкову позицію крана на одній з вершин графа  $G$ . На наступному кроці, використовуючи опис задачі про кран, наведений у [157], будемо шлях для задачі комівояжера на графі  $G$  з такою умовою: «Якщо з вершини виходять ребра і дуги, то по ребру можна рухатися лише в тому випадку, якщо не можна рухатися по дузі». Відповідно до підходу, наведеного в [157], перетворюємо задачу про кран в задачу комівояжера.

Задача про кран є відомою оптимізаційною задачею [158], яка зводиться до задачі комівояжера. У загальному вигляді задачу про кран формулюють так.

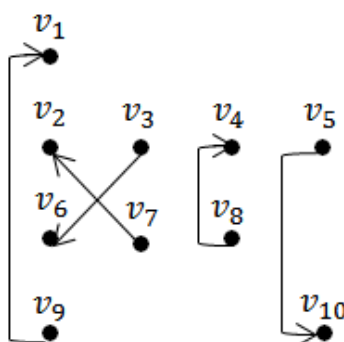
Нехай  $G=(F,E,A)$  — транспортна мережа з множиною вершин  $V$ , множиною ребер  $E$  та множиною дуг (орієнтованих ребер)  $A$ . Задано матрицю відстаней  $c_{ij}$ , яка визначена на ребрах та дугах мережі. Економічно дуги в мережі відповідають переміщенню вантажу з місця на місце, а ребра – холостим (незавантаженим) переміщенням крана. Необхідно побудувати такий маршрут крана, який проходив би через всі дуги мережі та мав би мінімальну довжину. Для зведення цієї задачі до задачі комівояжера прирівнюємо довжини всіх дуг мережі до нуля, а потім кожну пару вершин, які пов'язані дугою, об'єднуємо в одну вершину. Відстань між вершинами визначаємо так. Якщо вершина  $\bar{v}_i$  отримана об'єднанням вершин  $v_{i_1}$  та  $v_{i_2}$ , а вершина  $\bar{v}_j$  отримана об'єднанням вершин  $v_{j_1}$  та  $v_{j_2}$ , то  $\bar{c}_{ij} = \min(c_{i_1j_1}, c_{i_1j_2}, c_{i_2j_1}, c_{i_2j_2})$ . Розв'язавши задачу комівояжера з матрицею відстаней  $\bar{c}_{ij}$ , для отримання задачі про кран необхідно повернутися до початкової транспортної мережі та додати в маршрут комівояжера всі дуги.

Приклад перетворення задачі наведено в [159]. З цією метою прирівнюємо довжину всіх дуг  $d_i \in D, i = 1, \dots, m$ , графа  $G$  до нуля та об'єднуємо кожну пару вершин, об'єднаних дугою, в одну вершину, як запропоновано в [157]. Далі розв'язуємо задачу комівояжера з матрицею відстаней  $\bar{c}_{ij}$ , використовуючи один з добре відомих методів, наприклад, один з сучасних солверів (розв'язувачів) Concorde TSP solver (доступний за посиланням <http://www.math.uwaterloo.ca/tsp/concorde.html>).

Для отримання розв'язку задачі про кран, отже, і розв'язку РОСТУ повертаємося до початкового графа  $G$  та додаємо всі дуги з множини  $D$  у знайдений шлях комівояжера. Шлях переміщення, знайдений у результаті розв'язання задачі про кран на графі  $G$ , є розв'язком РОСТУ.

Приклад розв'язання РОСТУ. Відомо  $V = \{v_1, v_2, \dots, v_{10}\}$  – множина вершин, де  $v_1 = (1,0)$ ,  $v_2 = (1,1)$ ,  $v_3 = (2,1)$ ,  $v_4 = (3,1)$ ,  $v_5 = (4,1)$ ,  $v_6 = (1,2)$ ,  $v_7 = (2,2)$ ,  $v_8 = (3,2)$ ,  $v_9 = (1,3)$ ,  $v_{10} = (4,3)$ ;

$D = \{d_1, d_2, d_3, d_4, d_5\}$  – множина дуг, де  $d_1 = (v_9, v_1)$ ,  $d_2 = (v_3, v_6)$ ,  $d_3 = (v_7, v_2)$ ,  $d_4 = (v_8, v_4)$ ,  $d_5 = (v_5, v_{10})$ . На рис. 4.3, а показано множину вершин  $V$  та множину дуг  $D$ , які попарно об'єднують вершини з множини  $V$ ; на рис. 4.3, б – матрицю відстаней  $\|c_{ij}\|$ .



а

	$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$v_6$	$v_7$	$v_8$	$v_9$	$v_{10}$
$v_1$	0	1	2	3	4	2	3	4	100	6
$v_2$	1	0	2	3	3	1	100	3	2	5
$v_3$	2	2	0	1	2	2	1	2	3	4
$v_4$	3	3	1	0	1	3	2	100	4	3
$v_5$	4	3	2	1	0	4	3	2	5	2
$v_6$	2	1	100	3	4	0	1	2	1	4
$v_7$	3	2	1	2	3	1	0	1	2	3
$v_8$	4	3	2	1	2	2	1	0	3	2
$v_9$	3	2	3	4	5	1	2	3	0	3
$v_{10}$	6	5	4	3	100	4	3	2	3	0

б)

б

**Рис. 4.3.** Вершини та дуги графа (а), а також матриця відстаней  $\|c_{ij}\|$  (б)

Об'єднуємо вершини, зв'язані спільною дугою:  $\bar{v}_1 = v_9 \leftrightarrow v_1$ ;  $\bar{v}_2 = v_7 \leftrightarrow v_2$ ;  $\bar{v}_3 = v_3 \leftrightarrow v_6$ ;  $\bar{v}_4 = v_8 \leftrightarrow v_4$ ;  $\bar{v}_5 = v_5 \leftrightarrow v_{10}$ . Матрицю відстаней  $\|\bar{c}_{ij}\|$  для повного графа з вершинами  $\bar{V} = \{\bar{v}_1, \bar{v}_2, \bar{v}_3, \bar{v}_4, \bar{v}_5\}$  наведено на рис. 4.4. Для розв'язання задачі комівояжера з матрицею  $\|\bar{c}_{ij}\|$  використовуємо веб-сервіс та в результаті отримуємо замкнутий шлях комівояжера на вершинах  $\bar{V} = \{\bar{v}_1, \dots, \bar{v}_5\}$ :  $\bar{v}_4 \rightarrow \bar{v}_5 \rightarrow \bar{v}_3 \rightarrow \bar{v}_1 \rightarrow \bar{v}_2 \rightarrow \bar{v}_4$ .

	$\bar{v}_1$	$\bar{v}_2$	$\bar{v}_3$	$\bar{v}_4$	$\bar{v}_5$
$\bar{v}_1$	0	1	1	3	3
$\bar{v}_2$	1	0	1	1	3
$\bar{v}_3$	1	1	0	1	2
$\bar{v}_4$	3	1	1	0	1
$\bar{v}_5$	3	3	2	1	0

Рис. 4.4. Матриця відстаней  $\|\bar{c}_{ij}\|$

Для побудови шляху для порталного крана через усі вершини початкового графа  $G$  додаємо всі дуги з множини  $D$  в знайдений замкнутий шлях. У результаті перетворення отримуємо шлях порталного крана через всі вершини  $V = \{v_1, v_2, \dots, v_{10}\}$ , остаточний розв'язок для вхідної задачі виглядає так:  $v_8 \rightarrow v_4 \rightarrow v_5 \rightarrow v_{10} \rightarrow v_3 \rightarrow v_6 \rightarrow v_9 \rightarrow v_1 \rightarrow v_7 \rightarrow v_2 \rightarrow v_8$ .

4.6. Застосування задач завантаження контейнерів у логістичних процесах попередження та ліквідації надзвичайних ситуацій

Розповсюдженість та актуальність задач завантаження контейнерів визначають широкий спектр їх практичних застосувань. Розглянемо роль і місце задач завантаження контейнерів у системах попередження та ліквідації надзвичайних ситуацій.

В умовах сучасних ризиків техногенного і природного характеру, що призводять до надзвичайних ситуацій, актуальним є розроблення інформаційно-аналітичних систем попередження і усунення таких ситуацій.

До ефективності систем зазначеного класу має висуватися низка вимог, а саме: забезпечення мінімізації як людських, так і економічних втрат у разі усунення надзвичайних ситуацій (НС); за можливістю своєчасне попередження виникнення надзвичайних ситуацій у режимі моніторингу [160, 161].

Для кожної з можливих причин виникнення надзвичайної ситуації інформаційно-аналітична система має давати змогу вирішувати такі завдання:

- попередження виникнення надзвичайної ситуації;
- своєчасне реагування на надзвичайну ситуацію у разі її виникненні;
- формування ефективного плану усунення надзвичайної ситуації за рахунок координації всіх служб, сил і засобів;
- координація плану усунення наслідків НС.

У праці [144] запропоновано концепцію РІАП НС – Регіональної інформаційно-аналітичної підсистеми з надзвичайних ситуацій – інформаційно-аналітичної системи щодо запобігання та усунення наслідків НС у рамках регіону і країни в цілому (рис. 4.5).

Концепція передбачає оперативне забезпечення керівних органів виконавчої влади експертно-аналітичною, прогнозною, довідково-статистичною, фактографічною, контрольною-звітною та управлінською інформацією з використанням сучасних інформаційних технологій для розв'язання задач, пов'язаних з техногенною та екологічною безпекою та НС.

Одним із розділів інформаційно-аналітичної системи є підсистема «Планування заходів і підготовка рішень», орієнтована на розв'язання таких задач:

- підготовка проєктів рішень з планування заходів з ліквідації наслідків НС;
- формування експертної інформації на основі аналізу інформації з бази даних прецедентів за конкретними НС;
- визначення оцінки необхідних ресурсів і розроблення планів їх використання у разі ліквідації наслідків НС.

Надзвичайні ситуації, як потенційні, так і ті, що відбулись і потребують ліквідації, характеризуються необхідністю зосередження та транспортування великих обсягів матеріальних ресурсів в район події. Основним способом транспортування зазначених вантажів є контейнерні перевезення всіма доступними видами транспорту [160, 161]. Це призводить до необхідності розв'язання задач завантаження контейнерів на етапах планування заходів щодо постачання ресурсів та реагування на надзвичайну ситуацію, що

відбулась. Так, матеріальні ресурси, які заздалегідь зберігаються на складах на випадок виникнення НС, мають бути щільно упаковані у спеціальні контейнери та готові до транспортування.

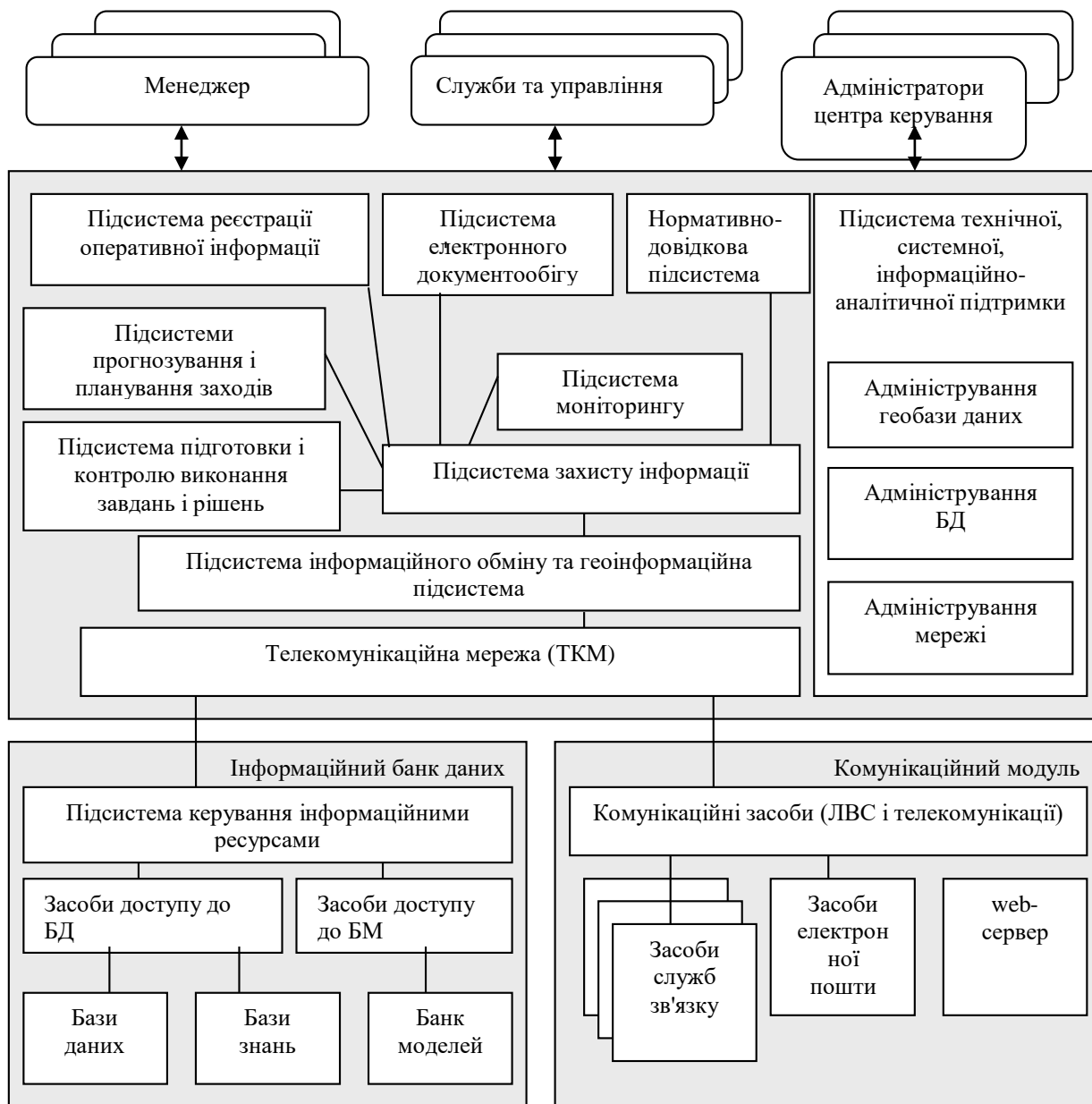


Рис. 4.5. Структурно-функціональна схема РІАП НС

У разі виникнення надзвичайних ситуацій постає задача пошуку найкоротших шляхів транспортування контейнерів з ресурсами за умов, коли частина шляхів може бути пошкодженою, що призведе до використання

різних видів транспорту. Отже, необхідно перевантажувати контейнери в процесі транспортування в район події.

Зазначене є підставою для використання запропонованих у монографії алгоритмів розв'язання задач завантаження контейнерів у підсистемі «Планування заходів і підготовка рішень» Регіональної інформаційно-аналітичної підсистеми з надзвичайних ситуацій РІАП НС.

Деякі результати, наведені в розділі, опубліковано в працях [42, 44, 48, 50, 52, 57].

## ВИСНОВКИ

У монографії розглянуто математичні моделі та методи розв'язання задач проєктування оптимального завантаження контейнерів з урахуванням геометричних обмежень та умов балансу.

Виконано системний аналіз задачі оптимального завантаження контейнерів у різних сферах застосування. У результаті зроблено такі висновки:

- контейнерні перевезення є підсистемою глобальної економічної та геополітичної системи, яка значно впливає на світові процеси;
- контейнерні перевезення мають тенденцію до кількісного зростання та технологічного розвитку на всіх рівнях – глобальному, регіональному та місцевому;
- оптимізацію завантаження контейнерів варто розглядати як на етапі виконання операцій переміщення контейнерів у контейнерному терміналі, так і на етапі завантаження окремих контейнерів, що зумовить суттєвий економічний ефект.

Розглянуто комбінаторні конфігурації, що описують комбінаторну структуру задачі оптимального балансного завантаження 3D об'єктів (вантажів) у контейнері, розділеному горизонтальними стелажми на підконтейнери.

Наведено математичну модель задачі оптимального балансного завантаження 3D об'єктів, в якій враховано не лише геометричні обмеження та баланс системи, а й комбінаторні особливості, пов'язані з генерацією розбиття множини об'єктів, що розміщені всередині підконтейнерів.

Описано математичну модель у вигляді задачі нелінійного програмування для оптимізації компоновки кластерів об'єктів (вантажів) у прямокутному контейнері, що дає змогу застосовувати сучасні НЛП-розв'язувачі для її розв'язання.

Запропоновано швидкий метод генерації допустимих стартових точок для пришвидчення процедури локальної оптимізації та поліпшення значення функції цілі.

Описано стратегії, моделі та методи розв'язання задач розміщення контейнерів у процесі їх перевантаження щодо побудови математичної моделі задачі розміщення контейнерів на залізничному терміналі та врахування геометричних та технологічних властивостей для її розв'язання, що дає змогу зменшити загальну вартість контейнерних операцій на залізничному терміналі.

## СПИСОК УМОВНИХ СКОРОЧЕНЬ

TEU – Twenty-foot Equivalent Unit – умовна одиниця вимірювання місткості контейнерів; заснована на об'ємі 20 футового (6,1 метра) інтермодального ISO-контейнера, що може транспортуватися різними видами транспорту: автомобільним, залізничним і морським

FEU – Forty-foot Equivalent Unit – умовна одиниця вимірювання місткості контейнерів; заснована на об'ємі 40-футового інтермодального ISO-контейнера

CBLP – комбінаторна задача балансного компоювання (Combinatorial Balance Layout Problem)

MBLP – багатокритеріальна задача балансного компоювання (Muticriteria Balance Layout Problem)

GLP – задача компоювання кластерів об'єктів (Group Layout Problem)

CPRTOP – задача розміщення контейнерів за умов експлуатації залізничного терміналу (Containers Placement In Rail Terminal Operations Problem)

TYSP – задача побудови розкладу обслуговування залізничного терміналу (Transshipment Yard Scheduling Problem)

НС – надзвичайна ситуація

## СПИСОК ЛІТЕРАТУРИ

1. Сергиенко И.В., Шило В.П. Задачи дискретной оптимизации: Проблемы, методы решения, исследования. НАН Украины; Институт кибернетики. К.: Наукова думка, 2003. 264 с.
2. Сергієнко І.В. Інформатика та комп'ютерні технології. Київ: Наук. думка, 2004. 432 с.
3. Сергієнко І.В. Методи оптимізації та системного аналізу для задач трансобчислювальної складності. К.: Академперіодика, 2010. 296 с.
4. М. З. Згуровский, А. А. Павлов. Труднорешаемые задачи комбинаторной оптимизации в планировании и принятии решений. Киев: Наукова думка, 2016.
5. Lyashko S.I. Generalized Optimal Control of Linear Systems with Distributed Parameters. Dordrecht, Boston, London. – Kluwer Academic Publishers. 2002. 455 p.
6. Klyushin D.A, Lyashko S.I., Nomirovsky D.A., Petunin Yu.,I., Semenov V.V. Generalized solutions of operator equations and extreme elements. Springer. 2012. New York, Dordrecht, Heidelberg, London. 200 p.
7. Hulianytskyi L. F., Riasna I. I. Formalization and classification of combinatorial optimization problems In: Optimization Methods and Applications (eds. Butenko S., Pardalos P. M., Shylo V.). Cham: Springer International Publishing AG, 2017. P. 239–250.
8. Lebedeva, T.T., Semenova, N.V. & Sergienko, T.I. Multi-Objective Optimization Problem: Stability against Perturbations of Input Data in Vector-Valued Criterion. *Cybern Syst Anal.* 2020, 56, pp. 953–958.
9. Sergienko I.V., Semenova N.V., Semenov V.V. Bilevel Optimization Problems of Distribution of Interbudgetary Transfers Under Given Limitations. *Cybern Syst Anal.* 2019, 55, pp. 905–913.
10. Семенова Н.В., Колечкіна Л.М. Векторні задачі дискретної оптимізації на комбінаторних множинах. Київ: Наукова думка, 2009.

11. Стецюк П.И. Приближенный метод эллипсоидов. Кибернетика и системный анализ. 2003. № 3. С. 141–146.
12. Шор Н.З., Журбенко Н.Г., Лиховид А.П., Стецюк П.И. Развитие алгоритмов недифференцируемой оптимизации и их приложения. Кибернетика и системный анализ. 2003. № 4. С. 80 – 94.
13. Крак Ю.В. Оптимізаційні постановки та методи в задачах дослідження маніпуляційних систем. Вісник Київського університету. Серія фізико-математичні науки. 2000. № 2. С. 56–62.
14. Грицик В.В., Шевченко А.І., Кісельова О.М. Математичні методи оптимізації та інтелектуальні комп'ютерні технології моделювання складних процесів і систем з урахуванням просторових форм об'єктів. Донецьк: ПП "Наука і освіта", 2011. 480 с.
15. Сергиенко И.В., Шило В.П. Современные подходы к решению сложных задач дискретной оптимизации. Проблемы управления и информатики. 2016. № 1. С. 32–40.
16. Бидюк П.И., Касицкий А.В. Анализ сходимости метода оптимизации Розенброка. Системні технології. 2011. № 3. С. 51–60.
17. Базилевич Р., Ждан А. Ієрархічна кластеризація складних схем. Вісн. Нац. ун-ту «Львів. політехніка». Комп'ют. науки та інформ. технології. 2008. № 616. С. 41–46.
18. Донець Г. П., Колєчкіна Л.М. Екстремальні задачі на комбінаторних конфігураціях. Полтава: ПУЕТ. 2001.
19. Стоян Ю.Г., Яковлев С.И. Математические модели и оптимизационные методы геометрического проектирования. Киев: Наукова думка. 1986. 268 с.
20. Элементы теории геометрического проектирования: монография. С. В. Яковлев, Н. И. Гиль, В.М. Комяк, И. В. Аристова и др.; под ред. В.Л.Рвачева. Киев: Наукова думка, 1995. 247 с.

21. Stoyan Y., Pankratov A., Romanova T. Placement Problems for Irregular Objects: Mathematical Modeling. Optimization and Applications. Springer, Cham. 2017, pp. 521–559.

22. Романова Т.Е. Средства построения математических моделей оптимизационных задач размещения геометрических объектов и их применение. Дис. д-ра техн. наук, 01.05.02. Харьков. 2003. 324 с.

23. Панкратов А. В. Математические модели, методы и информационные технологии решения оптимизационных задач размещения геометрических объектов. Дис. д-ра техн. наук, 01.05.02. Харьков. 2013. 334 с.

24. Chernov N., Stoyan Y., Romanova T. Mathematical model and efficient algorithms for object packing problem. Computational Geometry: Theory and Applications. 2010. Vol. 43, N 5. P. 535–553.

25. Chernov N., Stoyan Y., Pankratov A., Romanova T. Quasi-phi-functions and optimal packing of ellipses. Subm. to J. of Glob. Optim. 2014. Vol. 65. Issue 2. P. 283–307.

26. Стоян Ю.Г., Яськов Г.М., Романова Т.Є., Яковлев С.В. Пакування сферичних об'єктів: моделі, методи, застосування. Київ: Наукова думка, 2021. 280 с.

27. Stoyan Yu., Grebennik I., Romanova T., Kovalenko A. Optimized Packings in Space Engineering Applications: Part II. Chapter in contributed book "Modeling and Optimization in Space Engineering – State of the Art and New Challenges". Springer Optimization and its Applications, G. Fasano and J.Pintér (Editors), Springer, New York, 2019. P. 439–457.

28. Стоян Ю.Г., Семкин В.В., Чугай А.М. Оптимизация компоновки трехмерных объектов в многосвязной области с учетом кратчайших расстояний. Кибернетика и системный анализ. 2014. №3. С.58–70.

29. Grebennik I. V., Pankratov A. V., Chugay A. M., Baranov A. V. Packing n-dimensional parallelepipeds with the feasibility of changing their orthogonal orientation in an n-dimensional parallelepiped. Cybernetics and Systems Analysis. 2010. Vol. 46, N 5, pp. 793–802.

30. Стоян Ю. Г., Яковлев С. В., Пичугина О. С. Эвклидовы комбинаторные конфигурации: монографія. Харьков: Константа, 2017.
31. Яковлев С. В. Свойства задач комбинаторной оптимизации на полиэдрально-сферических множествах. Кибернетика и системный анализ. 2018. № 54(1). С. 111–123.
32. Яковлев С. В. О комбинаторной структуре задач оптимального размещения геометрических объектов. Доклады НАН Украины. 2017. № 9. С. 63–68.
33. Пичугина О. С., Яковлев С. В. Непрерывные функциональные представления в задачах дискретной оптимизации: монографія. Харьков: Золотая миля, 2018.
34. Yakovlev S. Convex extensions in combinatorial optimization and their applications. Optimization Methods and applications. Butenko S. et al. (Eds.). New York: Springer, 2017. P. 576–584.
35. Стоян Ю. Г., Ємець О. О. Теорія і методи евклідової комбінаторної оптимізації. Інститут системних досліджень освіти. Київ. 1993.
36. Стоян Ю.Г., Ємець О.О., Ємець Є.М. Оптимізація на полірозміщеннях: теорія та методи. Полтава: РВЦ ПУСКУ, 2005. 103 с.
37. Емец О., Роскладка А. О комбинаторной оптимизации в условиях неопределенности. Кибернетика и системный анализ. 2008. №44(5). С.35–44.
38. Гребенник И. В. Математические модели и методы комбинаторной оптимизации в геометрическом проектировании. Дис. д-ра техн. наук, 01.05.02. Харьков. 2006. 354 с.
39. Семенец В.В., Гребенник И.В., Листровой С.В., Минухин С.В., Овезгельдыев А.О. Модели и методы комбинаторной оптимизации в проектировании и управлении. Киев: Наукова думка, 2019. 176 с.
40. Литвиненко О.С. Методи генерації комбінаторних конфігурацій та їх застосування в математичному і комп'ютерному моделюванні задач перевезення та обробки вантажів. Дис. канд. техн. наук, 01.05.02, Харків, 2018.

41. Гребенник И.В., Коваленко А.А., Романова Т.Е., Урняева И.А., Шеховцов С.Б. Задача балансной компоновки цилиндрических объектов с учетом ее комбинаторных особенностей. *Біоніка інтелекту: науково-технічний журнал*. 2017. №1 (88). С.22–28.

42. Grebennik I., Dupas R., Lytvynenko O., Urniaieva I. Scheduling Freight Trains in Rail-rail Transshipment Yards with Train Arrangements. *International Journal of Intelligent Systems and Applications (IJISA)*, 2017. Vol.9, № 10, pp.12–19.

43. Гребенник И.В., Коваленко А.А., Романова Т.Е., Урняева И.А., Шеховцов С.Б. Комбинаторные конфигурации в оптимизационных задачах балансной компоновки. *Кібернетика та системний аналіз*. 2018. Т. 54, №2. С. 55–67.

44. Grebennik I., Khriapkin O., Ovezgeldyyev A., Pisklakova V., Urniaieva I. The Concept of a Regional Information-Analytical System for Emergency Situations. In: Murayama Y., Velev D., Zlateva P. (eds). *Information Technology in Disaster Risk Reduction. ITDRR 2017. IFIP Advances in Information and Communication Technology*, 2019. Vol 516. Springer, Cham.

45. Romanova T., Pankratov A., Litvinchev I., Pankratova Yu., Urniaieva I., Optimized Packing Clusters of Objects in a Rectangular Container. *Mathematical Problems in Engineering*. 2019. Vol. 2019, Article ID 4136430, 12 pages.

46. Romanova T., Litvinchev I., Grebennik I., Kovalenko A., Urniaieva I., Shekhovtsov S. Packing Convex 3D Objects with Special Geometric and Balancing Conditions. Chapter in contributed book "Intelligent Computing and Optimization", *Proceedings of the 2nd International Conference on Intelligent Computing and Optimization 2019 (ICO 2019)*. Springer Optimization and its Applications, Vasant, P., Zelinka, I., Weber, G.-W. (Editors), Springer, New York, 2020. P. 273–281.

47. Дубинський В.М., Лисенко Б.С., Каяфюк Р.В., Панкратов О.В., Романова Т.Є, Жмуд О.О. Опукла багатокутна оболонка для пари нерегулярних об'єктів. *Кібернетика та комп'ютерні технології*. 2021, №2. С. 13–24.

48. Romanova T., Pankratov A., Litvinchev I., Dubinskyi V., Infante L. Sparse Layout of Irregular 3D Clusters Journal of the Operational Research Society. 2021. Accepted.

49. Kalaida N., Dupas R., Grebennik I. (2021, April 21–23), “Optimization of order for containers placement schedule in rail terminal operations” [Conference session], PMS 2020/2021. 17th International Workshop on Project Management and Scheduling. <https://pms2020.sciencesconf.org/resource/page/id/15>

50. Grebennik I., Khriapkin O., Pisklakova V., Urniaieva I. Concept of a regional information-analytical system for emergency situations. 2nd IFIP Conference on Information Technology in Disaster Risk Reduction ITDRR 2017, October 25–27th, 2017, University of National and World Economy, Sofia, Bulgaria.

51. Grebennik, I., Kalaida, N., Urniaieva I., Ryabchenko, I., Reshetnik V. Placement of Different Types Containers at Railway Container Terminal Using Double-Stack (2021). Proc. 11th International Conference on Advanced Computer Information Technologies, ACIT 2021. Proceedings, 2020, Deggendorf, GERMANY, 15–17 September 2021.

52. Grebennik I., Dupas R., Lytvynenko O., Urniaieva I. Train arrangement in scheduling for Rail-rail transshipment yard. 7-th International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE – 2017), October 25–27th, 2017, University of National and World Economy Sofia, Bulgaria.

53. Grebennik I., Romanova T., Kovalenko A., Urniaieva I., Shekhovtsov S. Multicriteria balance layout problems of 3D-objects. 7-th International Conference on Application of Information and Communication Technology and Statistics in Economy and Education (ICAICTSEE – 2017), October 25–27th, 2017, University of National and World Economy Sofia, Bulgaria.

54. Romanova T., Stoyan Y., Pankratov A. et al. Sparsest balanced packing of irregular 3D objects in a cylindrical container. *European Journal of Operational Research*. 2021. V. 291(1). P. 84–100.

55. Grebennik I., Romanova T., Urniaieva I., Shekhovtsov S. Mathematical Model of Balanced Layout Problem Using Combinatorial Configurations. *Proc. International Conference on Advanced Computer Information Technology (ACIT 2018)*, June 1–3, 2018, Ceske Budejovice, Czech Republic. P. 18–21. Scopus.

56. Urniaieva I., Grebennik I., Romanova T., Pankratov A., Kovalenko A. Muticriteria Model of Balanced Layout Problem of 3D-Objects. *Proc. International Conference on Advanced Computer Information Technology (ACIT 2018)*, June 1–3, 2018, Ceske Budejovice, Czech Republic. P. 22–25.

57. Grebennik I., Dupas R., Urniaieva I., Kalaida N., Ivanov V. Mathematical Model of Containers Placement in Rail Terminal Operations Problem. *Proc. International Conference on Advanced Computer Information Technology (ACIT 2019, IEEE)*, June 5–7, 2019, Ceske Budejovice, Czech Republic. P. 129 –132.

58. Urniaieva I., Pankratov A., Romanova T., Grebennik I., Dupas R., Shekhovtsov S. Balance Packing Problem of Cuboids in an Optimized Cylindrical Container. *Proc. International Conference on Advanced Computer Information Technology (ACIT 2019, IEEE)*, June 5–7, 2019, Ceske Budejovice, Czech Republic. P. 133–136.

59. Container Handbook. German Insurance Association. 2006. [http://www.containerhandbuch.de/chb\\_e/index.html](http://www.containerhandbuch.de/chb_e/index.html)

60. Levinson, Marc *The Box: How the Shipping Container Made the World Smaller and the World Economy Bigger*. Princeton University Press. Retrieved 17 February 2013. <http://press.princeton.edu/chapters/s9383.html>

61. Container Supply Review (Report). World Shipping Council. May 2011. p. 1. Retrieved 18 July 2015. [https://www.worldshipping.org/public-statements/2011\\_container\\_supply\\_review\\_final.pdf](https://www.worldshipping.org/public-statements/2011_container_supply_review_final.pdf)

62. Pieriegud J. Analysis of the potential of the development of rail container transport market in Poland. Final report. European Commission, Directorate-

General for Regional and Urban Policy, European Union, 2019.  
[https://ec.europa.eu/regional\\_policy/sources/docgener/studies/pdf/report\\_pl\\_rail\\_container\\_transport\\_en.pdf](https://ec.europa.eu/regional_policy/sources/docgener/studies/pdf/report_pl_rail_container_transport_en.pdf)

63. Crainic T., Kim K. Intermodal transportation. Handbooks in operations research and management science. 2006. Vol. 14. 796 p.

64. Rodrigue J.-P. The Geography of Transport Systems. Fifth edition. New York: Routledge. 2020. 456 pages.

65. Steenken, D., Voß, S., Stahlbock, R. Container terminal operation and operations research – a classification and literature review/ OR Spectrum. 2004, 26, P. 3–49.

66. Port Economics, Management and Policy.  
<https://porteconomicsmanagement.org/pemp/contents/part3/container-terminal-design-equipment/>

67. Container Terminals: Facts and Figures. <https://container-xchange.com/blog/container-terminals/>

68. Мультимодальные и интермодальные пассажирские перевозки - разными видами транспорта. [Электронный ресурс]. 2017. Режим доступа: <http://wikitransport.com/transport/intermodal/>

69. <https://www.railfreight.com/intermodal/2019/03/05/fully-automated-rail-terminals-how-far-away/?gdpr=accept>

70. <https://www.reartek.com/klassifikaciya-konteinernih-termiinalov/>

71. <http://ukrspec.com/articles/containers.htm>

72. [https://www.boeing.com/resources/boeingdotcom/company/about\\_bca/pdf/CargoPalletsContainers.pdf](https://www.boeing.com/resources/boeingdotcom/company/about_bca/pdf/CargoPalletsContainers.pdf)

73. [https://www.airbus.com/content/dam/corporate-topics/publications/backgrounders/techdata/aircraft\\_characteristics/Airbus-Commercial-Aircraft-AC-A330.pdf](https://www.airbus.com/content/dam/corporate-topics/publications/backgrounders/techdata/aircraft_characteristics/Airbus-Commercial-Aircraft-AC-A330.pdf)

74. Modeling and Optimization in Space Engineering. Springer Optimization and Its Applications Editors G. Fasano and J.Pintér, Publisher Springer New York. 2012. Vol. 73.

75. Choong S. T., Cole M. H., Kutanoglu E. Empty container management for intermodal transportation networks. *Transportation Research Part E: Logistics and Transportation Review*. 2002. Vol. 38, № 6, pp. 423–438.

76. Tsamboulas D., Kapros S. Decision-Making Process in Intermodal Transportation. *Transportation Research Record: Journal of the Transportation Research Board*. 2000. Vol. 1707. pp. 86–93.

77. Bruns F., Knust S. Optimized load planning of trains in intermodal transportation. *OR Spectrum*. 2012. Vol. 34, №. 3, pp. 511–533.

78. Bontekoning Y., Macharis C., Trip J. Is a new applied transportation research field emerging? – A review of intermodal rail–truck freight transport literature. *Transportation Research Part A: Policy and Practice*. 2004. Vol. 38, N 1, pp. 1–34.

79. Falzarano A., Ketha S., Hawker J. S. Development of an intermodal network for freight transportation analysis. Rochester Institute of Technology, Accessed from <https://scholarworks.rit.edu/article/1206>

80. Bruns F., Goerigk M., Knust S., Schöbel A. Robust load planning of trains in intermodal transportation. *OR Spectrum*. 2014. Vol. 36, N 3. pp. 631–668.

81. Boysen N., Briskorn D., Knust S. Rail terminal operations. *OR Spectrum*. 2018. Vol. 40, N 2. pp. 317–318.

82. Guo P., Cheng W., Wang Y., Boysen N. Gantry crane scheduling in intermodal rail-road container terminals. *International Journal of Production Research*. Mar. 2018, pp. 1–18.

83. Pesch E., Boysen N., Fließner M., Jaehn F. A Survey on Container Processing in Railway Yards. *Transportation Science*. 2013. Vol. 47, N 3, pp. 312–329.

84. Pesch E., Boysen N., Jaehn F. Scheduling Freight Trains in Rail-Rail Transshipment Yards. *Transportation Science*. 2011. Vol. 45, N 2, pp. 199–211.

85. Boysen N., Jaehn F., Pesch E. New bounds and algorithms for the transshipment yard scheduling problem. *Journal of Scheduling*. 2012. Vol. 15, N 4, pp. 499–511.

86. Barketau M., Kopfer H., Pesch E. A Lagrangian lower bound for the container transshipment problem at a railway hub for a fast branch-and-bound algorithm. *Journal of the Operational Research Society*. 2013. Vol. 64, N 11, pp. 1614–1621.

87. Chernov N., Stoyan Yu., Romanova T., Pankratov A. Phi-Functions for 2D Objects Formed by Line Segments and Circular Arcs. *Advances in Operations Research*. 2012, vol. 2012, Article ID 346358, 26 pages.

88. Stoyan Y., Gil N., Romanova T., Scheithauer G. Phi-functions for complex 2D objects. *4OR Quarterly Journal of the Belgian, French and Italian Operations Research Societies*. 2004. Vol. 2(1). P. 69–84.

89. Bennell J., Scheithauer G., Stoyan Yu., Romanova T. Tools of mathematical modelling of arbitrary object packing problems. *Annals of Operations Research*, Publisher Springer Netherlands. 2010. Vol. 179I.1. P. 343–368.

90. Scheithauer G., Stoyan Yu., Romanova T. Mathematical modeling of interactions of primary geometric 3D objects. *Cybernet. Systems Anal.* 2005. Vol 41, N 3. P. 332–342.

91. Stoyan Y., Terno J., Gil M., Romanova T., Scheithauer G. Construction of a  $\Phi$ -function for two convex polytopes. *Applicationes Mathematicae*. 2002. Vol. 2, N 29. P. 199–218.

92. Стоян Ю.Г., Придатко Д.И., Романова Т.Е., Уварова М.А.  $\Phi$ -функции параллелепипедов и цилиндров. *Докл. НАН Украины*. 2002. № 10. С. 68–72.

93. Стоян Ю.Г. Чугай А.М. Построение свободной от радикалов  $\Phi$ -функции для шара и неориентированного многогранника. *Доповіді НАН України*. 2011. №12. С. 44–50.

94. Семкин В.В., Чугай А.М. Нормализованная  $\Phi$ -функция сферических сегментов. *Доповіді НАН України*. 2012. №12. С. 41–48.

95. Chernov N., Stoyan Y., Romanova T. Mathematical model and efficient algorithms for object packing problem. *Computational Geometry: Theory and Applications*. 2010. Vol. 43(5). P. 535–553.

96. Stoyan Yu., Yaskov G. Packing unequal circles into a strip of minimal length with jump algorithm. *Optimization Letters*. 2013. DOI10.1007/s 11590-013-0646-1, Online. ISSN 1862–4480.

97. Комяк В.М. Моделирование размещения геометрических объектов в областях произвольной пространственной формы. Геометричне та комп'ютерне моделювання. Харків: ХДАТОХ. 2002. Вип.1. С. 46–50.

98. Stoyan Yu., Romanova T., Zlotnik M., Scheithauer G., E.Stupak Packing of circles, rotating circular segments and polygons. *Проблемы машиностроения*. 2008. №1(11). С. 56–62.

99. Стоян Ю.Г., Семкин В.В., Чугай А.М. Оптимизация компоновки трехмерных объектов в многосвязной области с учетом кратчайших расстояний. *Кибернетика и системный анализ*. 2014. №3. С.58–70.

100. Стоян Ю.Г., Панкратов А.В., Романова Т.Е., Чернов Н.И. Квази-phi-функции для математического моделирования отношений геометрических объектов. *Доповіді НАН України*. 2014. Т.9. С. 49–54.

101. Dyckhoff H. A typology of cutting and packing problems. *European Journal of Operational Research*. 1990. Vol. 44. P. 145–159.

102. Wäscher G., Haußner H., Schumann H. An improved typology of cutting and packing problems. *European Journal of Operational Research*. 2007. Vol. 183. P. 1109–1130.

103. Sriramya P. A., Varthini P.B. State-of-the-Art Review of Bin Packing Techniques. *Eur. J. Scien. Res*. 2012. Vol. 86, N 3. P. 360–364.

104. <https://www.nefab.com/en/insights/load-optimization/>

105. Che C., Wang Y., Teng H. Test problems for quasi-satellite packing: Cylinders packing with behavior constraints and all the optimal solutions known. *Optimization Online* (2008).  
[http://www.optimizationonline.org/DB\\_HTML/2008/09/2093.html](http://www.optimizationonline.org/DB_HTML/2008/09/2093.html)

106. Birgin E. G., Martinez J. M., Ronconi D. P. Optimizing the packing of cylinders into a rectangular container. *European Journal on Operations Research*. 2005. Vol. 160, Issue 1. P. 19–33.

107. Egeblad J., Garavelli C., Lisi S., Pisinger D. Heuristics for Container Loading of Furniture. *European Journal of Operational Research*. 2010. Vol. 200, Issue 3. P. 881–892.

108. Fasano G. MIP-based heuristic for non-standard 3D-packing problems. *4OR: Quarterly Journal of the Belgian, French and Italian Operations Research Societies*. 2008. Vol. 6, Issue 3. P. 291–310.

109. Айгнер М. Комбинаторная теория. М.: Мир, 1982. 558 с.

110. Рейнгольд Э., Нивергельд Ю., Део Н. Комбинаторные алгоритмы. Теория и практика. М.: Мир, 1980. 478 с.

111. Сергиенко И.В. Математические модели и методы решения задач дискретной оптимизации. Киев: Наук. думка, 1988. 472 с.

112. Стоян Ю.Г., Яковлев С.В., Паршин О.В. Оптимизация квадратичных функций на множестве перестановок, отображенном в  $R^n$ . Докл. АН УССР. Сер. А. 1989. № 5. С.73–77.

113. Stoyan Yu., Yaskov G., Scheithauer G. Packing of various radii solid spheres into a parallelepiped. *Central European Journal of Operations Research*. 2003. Vol. 11, N 4. P. 389–407.

114. Коваленко А. А., Панкратов А. В., Романова Т. Е. Размещение объектов в контейнере параболоидной формы с круговыми стеллажами с учетом ограничений поведения. *Журнал обчислювальної та прикладної математики*. 2013. № 2 (112). С. 75–82.

115. Butenko S., Pardalos P., Shylo V. (eds) *Optimization Methods and Applications*. Springer Optimization and Its Applications, vol. 130. Springer, Cham.

116. Семкин В.В., Чугай А.М. Нормализованная Ф-функция сферических сегментов. *Доповіді НАН України*. 2012. №12. С.41–48.

117. Chazelle B, Edelsbrunner H, Guibas L.J. The complexity of cutting complexes. *Discrete & Computational Geometry*. 1989. 4(2). P. 139–81.

118. Fasano G., Pintér J. *Modeling and Optimization in Space Engineering*. Series: Springer Optimization and Its Applications. 2013. Vol. 73, XII. 404 p.

119. Fasano G., Pintér J. Space Engineering. Modeling and Optimization with Case Studies. Springer Optimization and its Applications, New York, 2016. Vol. 114, XV. 487 p.

120. Sun Z., Teng H. Optimal layout design of a satellite module. Engineering optimization. 2003. Vol. 35, N 5. P. 513–530.

121. Lei K. Constrained Layout Optimization Based on Adaptive Particle Swarm Optimizer. Advances in Computation and Intelligence. Series: Springer-Verlag Berlin Heidelberg. Zihua C., Zhenhua L., Zhuo K., Yong L. (Eds.). 2009. N 1. P. 434–442.

122. Stoyan Yu., Romanova T. Mathematical Models of Placement Optimization: Two- and Three-Dimensional Problems and Applications. Modeling and optimization in space engineering. Series: Springer optimization and its applications. Fasano G, Pinter J.D. (Eds.), XII. 2013. Vol. 73. P. 363–388.

123. Kovalenko A. A., Romanova T. E., Stetsyuk P. I. Balance layout problem for 3D-objects: mathematical model and solution methods. Cybernetics and Systems Analysis, 2015. Vol. 51, N 4, P. 556–565.

124. Stoyan Yu., Romanova T., Pankratov A., Kovalenko A., Stetsyuk P. Modeling and Optimization of Balance Layout Problems. Chapter in contributed book Space Engineering. Modeling and Optimization with Case Studies. Springer Optimization and its Applications, Editors G. Fasano and J.Pintér, New York, 2016. Vol. 114, XV, pp. 177–208.

125. Stoyan Yu. G., Grebennik I.V. Description and Generation of Combinatorial Sets Having Special Characteristics. International Journal of Biomedical Soft Computing and Human Sciences, Special Volume “Bilevel Programming, Optimization Methods, and Applications to Economics”. 2013. Vol. 18, N 1. P. 83–88.

126. Grebennik I., Lytvynenko O. Random generation of combinatorial sets with special properties. An International Quarterly Journal on Economics of Technology and Modelling Processes (ECONTECHMOD). 2016. Vol. 5, N 4, pp. 43–48.

127. Wachter A., Biegler L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*. 2006. Vol. 106, N 1, P. 25–57.

128. Stoyan, Yu., Romanova, T., Pankratov, A., Chugay, A. Optimized object packings using quasi-phi-functions. In: Fasano, G., Pinter, J. (eds.) *Optimized Packings with Applications*. Springer, New York. 2015, vol. 105, pp. 265–293.

129. Stoyan Yu., Pankratov A., Romanova T. Cutting and Packing problems for irregular objects with continuous rotations: mathematical modeling and nonlinear optimization. *Journal of the Operational Research Society*. 2016, Vol. 67(5), pp. 786–800.

130. Romanova, T., Bennell, J., Stoyan, Y., Pankratov, A.: Packing of concave polyhedra with continuous rotations using nonlinear optimization. *European Journal of Operational Research*. 2018. Vol. 268, pp. 37–53.

131. Stoyan Yu., Romanova T. Mathematical Models of Placement Optimisation: Two- and Three-Dimensional Problems and Applications. In: Fasano G., Pinter J. (eds.). *Modeling and Optimization in Space Engineering* Springer, New York. 2012, vol. 73, pp. 363–388.

132. Jingfa L., Gang L. Basin filling algorithm for the circular packing problem with equilibrium behavioural constraints. *SCIENCE CHINA Inf. Sci.*, 2010, Vol. 53(5), pp. 885–895.

133. Oliveira W.A., Moretti A.C., Salles-Neto L.L. A heuristic for the nonidentical circle packing problem. *Anais do CNMAC*. 2010, 3, pp. 626–632.

134. Xu Y.-C., Xiao R.-B., Amos M. A novel algorithm for the layout optimisation problem. *Proc. 2007 IEEE Congr. Evolut. Comput. (CEC07)*, IEEE Press, pp. 3938–3942.

135. Litvinchev I., Infante L., Ozuna L. Approximate packing: integer programming models, valid inequalities and nesting. In: Fasano G., Pinter J.D, (Eds.), *Optimized Packings and Their Applications (Ser.: Springer Optimization and Its Applications)*, 2015, vol. 105, pp. 117–135.

136. Pankratov A., Romanova T., Litvinchev I. Packing ellipses in an optimized rectangular container. *Wireless Networks*. 2018. doi.org/10.1007/s11276-018-1890-1.

137. Kallrath J., Rebennack S. Cutting ellipses from area-minimizing rectangles. *Journal of Global Optimization*, 2014, vol. 59, N 2, pp. 405–437.

138. Birgin E.G., Lobato R.D., Martinez J.M. Packing ellipsoids by nonlinear optimization. *Journal of Global Optimization*, 2016, vol. 65, N 4, pp. 709–743.

139. Stetsyuk P., Romanova T., Scheithauer G. On the global minimum in a balanced circular packing problem. *Optimisation Letters*, 2016, vol. 10, pp. 347–1360.

140. Zhanghua Fu., Wenqi Huang, Zhipeng Lü Iterated tabu search for the circular open dimension problem. *European Journal of Operational Research*. 2013, vol. 225, N 2, pp. 236–243.

141. Stoyan Y., Yaskov G., Packing equal circles into a circle with circular prohibited areas. *International Journal of Computer Mathematics*. 2012, vol. 89, N 10, pp.1355–1369.

142. Akeb H., Hifi M., Negre S. An augmented beam search-based algorithm for the circular open dimension problem. *Computers & Industrial Engineering*. 2011, vol. 61, N 2, pp. 373–381.

143. Litvinchev I., Infante L., Ozuna L. Packing circular like objects in a rectangular container. *Journal of Computer and Systems Sciences International*. 2015, vol. 54, N 2, pp. 259–267.

144. Peralta J., Andretta M., Oliveira J. Packing Circles and Irregular Polygons using Separation Lines. *Proceedings of the 7th International Conference on Operations Research and Enterprise Systems (ICORES 2018)*, 2018. pp. 71–77.

145. Kallrath J. Cutting circles and polygons from area minimizing rectangles. *Journal of Global Optimization*. 2009, vol. 43, pp. 299–328.

146. Alt H., Hurtado F. Packing Convex Polygons into Rectangular Boxes. *Lecture Notes in Computer Science*. 2000, vol. 2098, pp. 67–80, DOI: 10.1007/3-

540-47738-1\_5. Source Conference: Discrete and Computational Geometry, Japanese Conference, JCDCG 2000, Tokyo, Japan, November, 22–25, 2000.

147. Alt H., Berg M., Knauer C. Approximating minimum-area rectangular and convex containers for packing convex polygons. *Journal of Computational Geometry*. 2017, vol 8, N 1, pp. 1–10.

148. Kai Tang, Charlie Wang, Danny Z. Chen Minimum Area Convex Packing of Two Convex Polygons. *International Journal of Computational Geometry & Applications*. 2006, vol. 16, N 1, pp. 41–74.

149. Jones D. R. A fully general, exact algorithm for nesting irregular shapes. *Journal of Global Optimization*. 2013, vol. 59, pp.367–404.

150. Toledo F.M., Carravilla M.A., Ribeiro C., Oliveira J. F., Gomes A. M. The dotted board model: A new mip model for nesting irregular shapes. *International Journal of Production Economics*. 2013, vol. 145, N 2, pp. 478–487.

151. Bennell, J., Scheithauer, G., Stoyan, Y. et al. Optimal clustering of a pair of irregular objects. *J Glob Optim*. 2015, 61, pp. 497–524.

152. Bortfeldt A., Wäscher G. Constraints in container loading – A state-of-the-art review. *European Journal of Operational Research*. 2013, vol. 229, N 1, pp. 1–20.

153. Crainic T. Transportation Science Special Issue on Freight Transportation and Logistics, Part II. *Transportation Science*. 2016, vol. 50(4), pp. 1204–1205.

154. Yang W., Song H. Railway Container Terminal Station Layout and Operation Plan of Container Trucks. *LISS 2014: Proceedings of 4th International Conference on Logistics, Informatics and Service Science*, 2015, pp. 369–375.

155. Jaehn F., Otto A., Seifried K. Shunting operations at flat yards: retrieving freight railcars from storage tracks. *OR Spectrum: Quantitative Approaches in Management*, Springer. 2018, vol. 40(3), pp. 367–393.

156. Grebennik I., Lytvynenko O. Generating combinatorial sets with given properties. *Cybern. Syst. Anal.*, 2012, vol. 48(6), pp. 890– 898.

157. Melamed I. I., S. I. Sergeev, I. Kh. Sigal, 1989, The traveling salesman problem. Issues in theory, Autom. Remote Control, Vol. 50(9), pp. 1147–1173.
158. Avila, Thai & Corberán, Angel & Plana, Isaac & Sanchis, Jose Maria. (2014). The stacker crane problem and the directed general routing problem. Networks. Volume 65, Issue 1, January 2015, pp. 43–55.
159. Applegate D., Robert E. Bixby, Vasek Chvátal & William J. Cook. 2006. The Traveling Salesman Problem: A Computational Study, Princeton University Press.
160. Van De Walle B., Turoff M., Hiltz S. R. Information Systems for Emergency Management. 2014. Business & Economics. 424 p.
161. <http://ieeexplore.ieee.org/document/6601662>
162. Christensen S. G., Rouse D. M. Container Loading with multi-drop constraints. International translation in operational research. N 16. P. 727–743.
163. Bischoff E. E., Ja-nets F., Ratcliff M. S. W. Loading pallets with nonidentical items. European Journal of Operational Research. 1995. N 84. P.681–692.
164. Bischoff E. E., Ratcliff M. S. W. Loading multiple pallets. Journal of the Operational Research Society. 1995. N 46. P.1322–1336.
165. Bischoff E. E., Ratcliff M. S. W. Issues in the development of approaches to container loading. International Journal of Management Science. 1995. N 23 (4). P. 377–390.
166. Abdou G., Arghavani J. Interactive ILP Procedures for stacking optimization for the 3D pal-letization problem. International Journal of Production Re-search. 1997. N 35 (5). P. 1287–1304.
167. Gehring P., Bortfeld A. A genetic algorithm for solving the container loading problem. International Transaction in Operational Research. 1997. N 4 (5/6). P.401–418.
168. Bortfeld A., Wascher G. Container loading problems – a state-of-the-art review. Working Paper. 2012. N 7. P. 1–47.

169. Teng H., Sun S., Ge W., Zhong W. Layout optimization for dishes installed on a rotating table – the pack-ing problem with equilibrium behavioral constraints. *Science in China. Series A – Mathematics Physics Astronomy & Technological Sciences*. 1994. N 37 (10). P. 1272–1280.

170. Sun Z., Teng H. Optimal layout design of a satellite module. *Engineering optimization*. 2003. N 35 (5). P. 513–530.

171. Chazelle B., Edelsbrunner H., Guibas L. J. The complexity of cutting complexes. *Discrete & Computational Geometry*. 1989. N 4, (2). P. 139–181.

172. Teng H., Che C. , Wang Y. , Chen Y. Test problems of circles in circle packing with constraints and known the optimal solutions. *Optimization Online* (2004). URL: [http://www.optimization-online.org/DB\\_FILE/2004/10/976.pdf](http://www.optimization-online.org/DB_FILE/2004/10/976.pdf)

1

Xu. Y.-C. Xiao R.-B., Amos M. A Novel Genetic Algorithm for the Layout Optimization Problem. *Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC07)*, IEEE Press. 2007. P. 3938–3942.

175. Lei K. *Constrained Layout Optimization Based on Adaptive Particle Swarm Optimizer*. *Advances in Computation and Intelligence*. Series: Springer-Verlag Berlin Heidelberg. Zhihua C., Zhenhua L., Zhuo K., Yong L. (Eds.). 2009. N 1. P. 434– 442.

176. Oliveira W. A., Salles Neto L. L., Moretti A. C., Reis E. F. Nonidentical circle packing problem: multiple disks installed in a rotating circular container. *International Transactions in Operational Research*. 2014. P. 1–16. URL: <http://arxiv.org/ftp/arxiv/papers/1401/1401.4952.pdf>

177. Zhou C., Gao L., Gao H. Particle swarm optimization based algorithm for constrained layout optimization. *Control and Decision*. 2005. N 20 (1). P. 36–40.

178. Xiao R. B., Xu Yi C., Amos M. Two hybrid compaction algorithms for the layout optimization problem. *BioSystems*. 2006. N 90. P. 560–567.

179. Li N., Liu F., Sun D. B. A study on the particle swarm optimization with mutation operator constrained layout optimization. *Chinese J Comput.* 2004. N 27. P. 897–903.

180. Lei K. Y., Qiu Y. H. A study of constrained layout optimization using adaptive particle swarm optimizer. *J. Comput Research Develop.* 2006. N 43. P. 1724–1731.

181. Shi Y., Eberhart R. C. A modified particle swarm optimizer. In *Proc. of the IEEE Con. Evolutionary Computation*, IEEE Computer Press, Piscataway. 1998. P. 69–73.

182. Ning L., Fei L., Debao S. A study on the particle swarm optimization with mutation operator constrained layout optimization. *Chinese Journal of Computers.* 2004. N 27(7). P. 8897–9039.

183. Fei T., Teng H. F. A modified genetic algorithm and its application to layout optimization. *Journal of Software.* 1999. N 10. P. 1096–1102.

184. Qian Z. Q., Teng H. F., Sun Z. G. Human-computer interactive genetic algorithm and its application to constrained layout optimization. *Chinese J Comput.* 2001. N 24. P. 553–559.

185. Qian, Z. Q., Teng H. F., Xiong D. L. et al. Human-computer cooperation genetic algorithm and its application to complex layout design. In *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning*, Singapore. 2002. N 1. P. 299–302.

186. Sun Z. G., Qian Z. Q., Teng H. F. A hybrid genetic algorithm and its application to complex layout optimization. In *Proceeding of 4th World Congress of Structural and Multidisciplinary Optimization*, DUT Printing House, Dalian, China. 2002. P. 51–56.

187. Xu Y.-C., Dong F.-M., Liu Y., Xiao R.-B., Amos M. Ant Colony Algorithm for the Weighted Item Layout Optimization Problem. 2010. URL: <http://arxiv.org/ftp/arxiv/papers/1001/1001.4099.pdf>

188. Dorigo M., Caro G. D. Ant colony optimization: A new meta-heuristic. *Proceeding of 1999 Congresson Evolutionary Computation.* 1999. P. 1470–1477.

189. Oliveira W. A., Salles Neto L. L., Moretti A. C. A heuristic for the nonidentical circle packing problem. *Anais do CNMAC*. 2010. N 3. P. 626–632.

190. Xu. Y.-C., Xiao R.-B., Amos M. A Novel Genetic Algorithm for the Layout Optimization Problem. *Proceedings of the 2007 IEEE Congress on Evolutionary Computation (CEC07)*, IEEE Press. 2007. P. 3938–3942.

191. Huang W. Q., Chen M. Note on: an improved algorithm for the packing of unequal circles within a larger containing circle. *Comput. Indust. Eng.* 2006. N 50. P. 338–344.

192. Wang H. Q., Huang W. Q., Zhang Q. et al. An improved algorithm for the packing of unequal circles within a larger containing circle. *Eur. J. Oper. Res.* 2002. N 141. P. 440–453.

193. Wang Y. S., Shi Y. J., Teng H. F. An improved scatter search for circles packing problem with the equilibrium constraint. *Chinese J. Comput.* 2009. N 32. P. 1214–1221.

194. Wen Q. H., Yu L., Chu M. L., Ru C. X. New heuristics for packing unequal circles into a circular container. *Computers & Operations Research*. 2006. N 33(8). P. 2125–2142.

195. Fasano G., Pinte'r J. D. (Eds.) *Model Development and Optimization for Space Engineering: Concepts, Tools, Applications, and Perspectives*. Modeling and optimization in space engineering. Series: Springer optimization and its applications. 2013. Vol. 73. P. 1–31.

196. Fasano G. A Mixed Integer Linear Programming Model for Traffic Logistics Management at the International Space Station. Modeling and optimization in space engineering. Series: Springer optimization and its applications. 2013. Vol. 73. P. 215–233.

197. Junqueira L., Morabito R., Yamashita D. S., Yanasse H.H. Optimization Models for the Three-Dimensional Container Loading Problem with Practical Constraints. Modeling and optimization in space engineering. Series: Springer optimization and its applications. 2013. Vol. 73. P. 271–294.

198. Stoyan Yu., Romanova T. *Mathematical Models of Placement Optimization: Two- and Three-Dimensional Problems and Applications. Modeling and optimization in space engineering. Series: Springer optimization and its applications.* 2013. Vol. 73. P. 363–388.

199. Fadel G. M., Wiecek M. M. *Packing Optimization of Free-Form Objects in Engineering Design. Optimized Packing's with Applications.* Springer

,

,

2

0

2

203. Litvinchev I., Infante L., Ozuna L. *Approximate Packing: Integer Programming Models, Valid Inequalities and Nesting.* *Optimized Packing's with* 205.

1

204. Стоян Ю. Г., Гиль Н. И. Задачи размещения объектов с учетом их центров тяжести. В кн. *Методы и алгоритмы размещения плоских геометрических объектов.* Киев: Наукова думка, 1976. С. 230–238.

205. Стоян Ю. Г., Пономаренко Л. Д. Сумма Минковского и годограф вектор-функции плотного размещения. Докл. АН УССР. Сер. А, 1977, № 10. С. 888–890.

206. Стоян Ю. Г., Кулиш Е. Н. *Автоматизация проектирования компоновки оборудования летательных аппаратов.* М.: Машиностроение, 1984.

207. Stetsyuk P., Romanova T., Scheithauer G. On the global minimum in a balanced circular packing problem. *Optim. Letters.* 2015. DOI 10.1007/s11590-015-0937-9.

208. Shor N.Z. *Nondifferentiable Optimization and Polynomial Problems.* Boston Dordrecht. London: Kluwer Academic Publishers, 1998. 412 p.

e

1

1

209. Preparata F.P., Shamos M.I. *Computational Geometry: An Introduction*. Springer. 1985. 400 p. doi.org/10.1007/978-1-4612-1098-6.
210. Avis D., Bremner D., Seidel R. How good are convex hull algorithms? *Computational Geometry: Theory and Applications*. 1997. 7 (5–6). P. 265–301. doi.org/10.1016/S0925-7721(96)00023-5.
211. Cormen T.H., Leiserson C.E., Ronald L. Rivest R.L., Stein C. *Introduction to Algorithms, Second Edition*. Section 33.3: Finding the convex hull. MIT Press and McGraw-Hill. 2001. P. 947–957. ISBN 0-262-03293-7.
212. De Berg M., Cheong O., Van Kreveld M., Overmars M. *Computational Geometry Algorithms and Applications*. Berlin: Springer. 2008. P. 2–14. doi:10.1007/978-3-540-77974-2.
213. Scheithauer G. *Introduction to Cutting and Packing Optimization. Problems, Modeling Approaches. Solution Methods*. Springer. 2018. 410 p. ISBN 978-3-319-64403-5.
214. Alt H., de Berg M., Knauer C. Approximating Minimum-Area Rectangular and Convex Containers for Packing Convex Polygons. In: Bansal N., Finocchi I. (eds). *Algorithms. ESA 2015. Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg. 2015. 9294. P. 25–34. doi.org/10.1007/978-3-662-48350-3\_3.
215. Yagiura M., Umetani S., Imahori S. *Cutting and Packing Problems From the Perspective of Combinatorial Optimization*. Springer. 2021. 300 p. ISBN 978-4-431-55291-8.
216. Tang K., Wang C.C.L., Chen D.Z. Minimum area convex packing of two convex polygons. *International Journal of Computational Geometry & Applications*. 2006. 16 (1). P. 41–74. doi.org/10.1142/S0218195906001926.
217. Kallrath J. Cutting Circles and Polygons from Area-Minimizing Rectangles. *Journal of Global Optimization*. 2009. 43. P. 299–328. doi.org/10.1007/s10898-007-9274-6.

218. Ahn H.K., Cheong O. Aligning Two Convex Figures to Minimize Area or Perimeter. *Algorithmica*. 2012. 62. P. 464–479. doi.org/10.1007/s00453-010-9466-1.

219. Park D., Bae S.W., Alt H., Ahn H.K. Bundling three convex polygons to minimize area or perimeter. *Computational Geometry*. 2016. 51. P. 1–14. <https://doi.org/10.1016/j.comgeo.2015.10.003>.

220. Kallrath J., Frey M.M. Packing Circles into Perimeter-Minimizing Convex Hulls. *Journal of Global Optimization*. 2019. 73 (4). P. 723–759. doi.org/10.1007/s10898-018-0724-0.

221. Kallrath J., Frey M.M. Minimal surface convex hulls of spheres. *Vietnam J. Math.* 2018. 46. P. 883–913. doi.org/10.1007/s10013-018-0317-8.

222. Chernov N., Stoyan Yu, Romanova T. Mathematical model and efficient algorithms for object packing problem. *Computational Geometry*. 2010. 43 (5). P. 535–553. doi.org/10.1016/j.comgeo.2009.12.003.

223. Stoyan Y., Pankratov A., Romanova T. Placement Problems for Irregular Objects: Mathematical Modeling, Optimization and Applications. In: Butenko S., Pardalos P., Shylo V. (eds). *Optimization Methods and Applications. Springer Optimization and Its Applications*. 2017. 130. P. 521–559. Springer, Cham. doi.org/10.1007/978-3-319-68640-0\_25.

224. Chernov N., Stoyan Y., Romanova T., Pankratov A. Phi-functions for 2D objects formed by line segments and circular arcs. *Advances in Operations Research*. 2012. doi.org/10.1155/2012/346358.

225. Stoyan Yu., Pankratov A., Romanova T. Cutting and Packing problems for irregular objects with continuous rotations: mathematical modeling and nonlinear optimization. *J. Oper. Res. Soc.* 2016. 67 (5). P. 786–800. <https://doi.org/10.1057/jors.2015.94>.

226. Wächter A., Biegler L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*. 2006. 106 (1). P. 25–57. doi.org/10.1007/s10107-004-0559-y.

227. Bortfeldt, A., Wäscher, G. Constraints in container loading – A state-of-the-art review. *European Journal of Operational Research* 229(1), 1–20 (2013).
228. Stoyan Yu., Romanova, T. Mathematical Models of Placement Optimization: Two- and Three-Dimensional Problems and Applications. In: Fasano G., Pinter J.D. (Eds.). *Modeling and Optimization in Space Engineering*, (Ser. Springer Optimization and Its Applications), vol. 73, pp. 363-388 (2012).
229. Chernov N., Stoyan Y., Romanova T., Pankratov A. Phi-Functions for 2D Objects Formed by Line Segments and Circular Arcs. *Advances in Operations Research*, doi:10.1155/2012/346358 (2012).
230. Romanova T., Bennell J., Stoyan Y., Pankratov A. Packing of concave polyhedra with continuous rotations using nonlinear optimization. *European Journal of Operational Research*, 268, (37–53), (2018).
231. Wachter A., Biegler L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming* 106(1), 25–57, (2006).
232. Dotoli M., Epicoco N., Falagario M., Palma D., and Turchiano B., A Train Load Planning Optimization Model for Intermodal Freight Transport Terminals: A Case Study. 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013. <https://doi.org/10.1109/smc.2013.613>
233. Vis I. F. A. and de Koster R., Transshipment of containers at a container terminal: An overview. *European Journal of Operational Research*, vol. 147, N 1, pp. 1–16, 2003. [https://doi.org/10.1016/s0377-2217\(02\)00293-x](https://doi.org/10.1016/s0377-2217(02)00293-x)
234. Perrault W. Load Sequencing for Double-Stack Trains. Université de Montréal, M.Sc. dissertation, 2018.
235. Mantovani S., Morganti G., Umang N., Crainic T. G., Frejinger E., and Larsen E. The load planning problem for double-stack intermodal trains. *European Journal of Operational Research*, vol.267(1), pp. 107–119, 2018. <https://doi.org/10.1016/j.ejor.2017.11.016>

*Наукове видання*

**НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ  
ІНСТИТУТ ПРОБЛЕМ МАШИНОБУДУВАННЯ ім. А.М. ПІДГОРНОГО**

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ**

**ГРЕБЕННІК ІГОР ВАЛЕРІЙОВИЧ  
РОМАНОВА ТЕТЯНА ЄВГЕНІВНА  
УРНЯЄВА ІННА АНАТОЛІЇВНА  
ДУБІНСЬКИЙ ВОЛОДИМИР МАКСИМОВИЧ  
КАЛАЙДА НАДІЯ СТАНІСЛАВІВНА**

**ОПТИМІЗАЦІЯ БАЛАНСНОГО І КЛАСТЕРНОГО  
КОМПОНУВАННЯ ТА ЗАВАНТАЖЕННЯ КОНТЕЙНЕРІВ**

Київ, Науково-виробниче підприємство  
«Видавництво “Наукова думка” НАН України», 2021

Підп. до друку 00.00.2021. Формат 60×90/16. Папір офс.  
Гарн. Таймс. Друк офс. Ум. друк. арк. 12,00. Обл.-вид. арк.14,80.  
Тираж 000 прим. Зам. № 00-00