НАЦІОНАЛЬНА АКАДЕМІЯ НАУК УКРАЇНИ

ІНСТИТУТ КІБЕРНЕТИКИ ІМЕНІ В.М. ГЛУШКОВА

Кваліфікаційна наукова

праця на правах рукопису

Рибальченко Олег Валерійович

УДК 519.8

ДИСЕРТАЦІЯ

МОДЕЛІ ТА МЕТОДИ МАРШРУТИЗАЦІЇ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

Спеціальність 113 – «Прикладна математика»

Галузь знань 11 – «Математика та статистика»

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело
Рибальченко О.В.

Науковий керівник: Гуляницький Леонід Федорович

доктор технічних наук,

член-кореспондент НАН України

Київ – 2023

АНОТАЦІЯ

Рибальченко О.В. Моделі та методи маршрутизації безпілотних літальних апаратів. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 113 "Прикладна математика". – Інститут кібернетики імені В.М. Глушкова Національної академії наук України, Київ. – 2023.

Зміст дисертації. У **вступі** обґрунтовано актуальність теми, сформульовано мету та задачі досліджень, розкрито наукову новизну та практичну цінність роботи, представлено її загальну характеристику.

У **розділі 1** наведено огляд літератури за темою дисертації, сформульовано необхідні означення, які у подальшому використовуються при формулюванні результатів дисертації.

У **розділі 2** наведено змістовні постановки двох запропонованих задач визначено їх ключові елементи, мету, початкові умови та існуючі обмеження. Розглянуто основні проблеми задачі планування місії гібридної системи та задачі планування місії команди БПЛА.

У **розділі 3** задачу планування місії гібридної системи та задачу планування місії команди БПЛА формалізовано за допомогою побудови двох відповідних математичних моделей. Описано та формалізовано початкові умови, обмеження, цільові функції для обох задач.

У **розділі 4** наведено обґрунтування вибору основних методів розв'язування поданих задач. Для кожної із задач описано розроблені жадібний алгоритм, алгоритм детермінованого локального пошуку, алгоритм оптимізації мурашиними колоніями. Формалізовано процес вибору задіяних баз на маршруті для задачі планування місії гібридної системи та процес вибору початкового розміщення БПЛА для задачі планування місії команди БПЛА.

У **розділі 5** наведено опис розробленого обчислювального блоку, протоколів обміну даними як всередині блоку, так і з зовнішніми системами, обґрунтовано застосовані технічні засоби та рішення.

У **розділі 6** подано результати чисельного експерименту та аналіз ефективності обраних алгоритмів для обох наведених задач.

Ключові слова: безпілотні літальні апарати, планування місій, гібридні системи, оптимізація маршрутів, математичне моделювання, оптимізація мурашиними колоніями, логістика

ABSTRACT

Rybalchenko O.V. Models and methods of routing unmanned aerial vehicles. - Qualification scientific work with the manuscript copyright.

Dissertation for a Doctor of Philosophy Degree by specialty 113 Applied Mathematics – V.M. Glushkov Institute of Cybernetics of the National Academy of Science of Ukraine. – Kyiv, 2023.

Contents of the dissertation. The **introduction** substantiates the relevance of the topic, formulates the purpose and objectives of the research, reveals the scientific novelty and practical value of the work, and presents its general characteristics.

Chapter 1 provides a publications review on the topic of the thesis, formulates the necessary definitions that are subsequently used in formulating the results of the thesis.

Chapter 2 presents meaningful formulations of two proposed problems, defines their key elements, purpose, initial conditions, and existing constraints. The main aspects of the hybrid system mission planning problem and the UAV team mission planning problem are considered.

Chapter 3 formalizes the hybrid system mission planning problem and the UAV team mission planning problem by constructing two corresponding mathematical models. Initial conditions, constraints, objective functions for both problems are formalized and described.

Chapter 4 provides a rationale for the choice of the main methods for solving given problems. For each of the problems, a developed greedy algorithm, deterministic local search algorithm, and ant colony optimization algorithm are described. The process of selecting involved bases on the route for the hybrid system

mission planning problem and the process of selecting the initial placement of UAVs for the UAV team mission planning problem are formalized.

Chapter 5 describes the developed computational unit, data exchange protocols both within the unit and with external systems, substantiates the applied technical means and solutions.

Chapter 6 presents the results of a numerical experiment and analyzes the effectiveness of selected algorithms for both presented problems.

Keywords: unmanned aerial vehicles, mission planning, hybrid systems, route optimization, mathematical modeling, ant colony optimization, logistics

Список публікацій здобувача

Публікації, в яких опубліковано основні наукові результати дисертації

1. Гуляницький Л.Ф., Рибальченко О.В. Оптимізація маршрутів при плануванні місій гібридних транспортних систем "Дрон+Транспортний засіб". *Cybernetics and Computer Technologies*. *2023*. **3**. С. 44–58. https://doi.org/10.34229/2707-451X.23.3.4
2. Гуляницький Л.Ф., Рибальченко О.В. Формалізація проблеми оптимізації місць базування та маршрутів групи БПЛА. *Cybernetics and Computer Technologies. 2021*. **4**. С. 12–26. https://doi.org/10.34229/2707-451X.21.4.2
3. Гуляницький Л.Ф., Рибальченко О.В. Формалізація та розв'язування одного типу задач маршрутизації БПЛА. *Теорія оптимальних рішень*. *2018*. **17.** С. 107-114. http://dspace.nbuv.gov.ua/handle/123456789/144979
4. Hulianytskyi L. Rybalchenko O. Optimization of decisions when planning a UAV group mission with alternative depots. *International Scientific Symposium «Intelligent Solutions», September 27-28, 2023, Kyiv, Ukraine*. 2023. P.245-256. https://ceur-ws.org/Vol-3538/Paper_22.pdf

ЗМІСТ

ВСТУП.....8

РОЗДІЛ 1. ОГЛЯД РЕЛЕВАНТНИХ ДОСЛІДЖЕНЬ 15

1.1 Про задачі планування місій з використанням БПЛА.....15

1.2 Огляд та класифікація основних класів наближених методів комбінаторної оптимізації 20

1.3 Висновки до розділу 27

РОЗДІЛ 2. ЗМІСТОВНІ ПОСТАНОВКИ ЗАДАЧ.....28

2.1 Вступні положення.....28

2.2 Про основні проблеми планування місій з гібридними системами 28

2.3 Змістовні постановки проблем оптимізації місій гібридної системи 30

2.4 Змістовна постановка проблеми планування місій команди БПЛА 32

2.5 Висновки до розділу 34

РОЗДІЛ 3. МАТЕМАТИЧНІ МОДЕЛІ ЗАДАЧ.....36

3.1 Вступні положення.....36

3.2 Математична модель задачі планування місій гібридних комплексів.....38

3.3 Математична модель задачі планування командних місій БПЛА.....42

3.4 Висновки до розділу 44

РОЗДІЛ 4. РОЗРОБКА АЛГОРИТМІВ ПЛАНУВАННЯ БЕЗПІЛОТНИХ МІСІЙ.....45

4.1 Вступні положення.....45

4.2 Основні поняття та позначення 46

4.3 Визначення початкового розміщення БПЛА.....48

4.4 Ключові аспекти реалізованого жадібного алгоритму.....49

4.5 Загальна схема алгоритму детермінованого локального пошуку 50

4.6 Ключові аспекти реалізованого алгоритму ДЛП 51

4.7 Загальна схема алгоритму ОМК для задачі планування командних місій. 53

4.8 Схема алгоритму ОМК для задачі планування місій гібридних систем.....62

4.9 Висновки до розділу 66
РОЗДІЛ 5. ПИТАННЯ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 68
5.1 Вступні положення....68
5.2 Структура програмного комплексу планувальника місій....69
5.3 Інтеграція обчислювального блоку з системами керування БПЛА 77
5.3.1 UgCS (Universal Ground Control Software) 78
5.3.2 Mission Planner 78
5.3.3 APM Planner 78
5.4 Висновки до розділу 78
РОЗДІЛ 6. РЕЗУЛЬТАТИ ЧИСЕЛЬНИХ ЕКСПЕРИМЕНТІВ 80
6.1 Результати дослідження оптимізації плану місій з гібридними системами...80
6.2 Результати дослідження алгоритмів оптимізації плану командних місій.....82
6.3 Аналіз результатів чисельних експериментів та висновки 90
РОЗДІЛ 7. ВИСНОВКИ....91
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 94
ДОДАТОК ГРАФІЧНИЙ МАТЕРІАЛ 103

ВСТУП

Протягом останніх десятиліть вивчення проблем планування місій дронів, перш за все, безпілотних літальних апаратів (БПЛА) стало сферою інтенсивних досліджень з великою кількістю застосувань. Прямо чи опосередковано БПЛА використовуються при моніторингу, спостереженнях, доставці ліків і товарів, проведенні пошуково-рятувальних робіт у зоні стихійного лиха, організації мобільних мереж, зборі даних; окремим напрямом є мілітарні застосування [1-10]. При такому плануванні виникають проблеми оптимізації, від розв'язування яких у межах обмеженого часу планування залежить ефективність виконання поставлених завдань.

Дослідження і розв'язування багатьох таких оптимізаційних проблем є розвитком тематики, яка стосується задач маршрутизації транспортних засобів (ТЗ) (Vehicle Routing Problem, VRP) [11]. Специфіка прикладних задач, які розглядаються, вносить корективи до тлумачення як терміну "вартість перевезень", так і "транспортний засіб" (ТЗ). Особливий клас задач породжують ситуації, у яких БПЛА можуть стартувати із декількох, зокрема рухомих пунктів базування та фінішувати в низці цих чи інших пунктів – такі задачі актуальні, перш за все, у логістиці і військовій сфері. При оптимізації маршрутів у цьому випадку вантажопідйомність БПЛА може не грати ключову роль (наприклад, при аеророзвідці), але більш суттєве значення мають обмеження на ресурс дрона – обсяг пального чи ємність акумулятора.

Велике значення в області маршрутизації БПЛА має задача планування місій, коли кілька БПЛА використовують різноманітні пункти (депо, бази) для старту та посадки. Ці бази можуть слугувати як основними місцями дислокації, так і станціями обслуговування БПЛА. Існують випадки фіксованого розташування баз – стаціонарні (коли кожен БПЛА закріплений за конкретною базою), альтернативні (де деякі можуть бути стартовими або фінішними) та динамічні – коли вони розміщені вздовж маршруту рухомого транспортного засобу [12, 13, 14].

Актуальність теми. Нині поширюється тенденція до розширення сфер використання безпілотних апаратів (дронів), зокрема БПЛА. Світовий досвід розвитку безпілотної авіації, насамперед, у таких розвинених країнах, як США, Німеччина, Ізраїль, показує, що БПЛА в найближчі 10–15 років зможуть виконувати більшість завдань, які вирішуються сьогодні пілотованими засобами. Створення і застосування перспективних систем цивільної та військової техніки обумовлюють нагальну необхідність розробок у сфері математичних методів підтримки прийняття та оптимізації рішень. Застосування БПЛА у різних сферах потребує розробки низки нових математичних моделей руху транспортних засобів та розробки алгоритмів планування операцій і адаптивної побудови оптимальних маршрутів при обстеженні чи/та обслуговуванні заданої множини об'єктів.

У роботі розглядаються і досліджуються питання оптимізації рішень при плануванні логістичних, моніторингових чи розвідувальних місій при виконанні завдання, яке полягає в обстеженні та/або обслуговуванні заданої множини цілей (клієнтів, об'єктів). При цьому розглядаються два класи таких задач: коли групи БПЛА або інших мобільних роботизованих систем працюють у команді і враховується можливість старту та фінішу на альтернативних базах і коли завдання виконується одним дроном (у більшості випадків БПЛА), який може транспортуватися спеціальним транспортом, що рухається по заданому маршруту.

Важливість дослідження цієї проблематики в Україні підкреслюють як хід бойових дій, так і обсяги тендерних закупівель від державних структур та військових частин [15, 16].

При формалізації виділених класів проблем виникають складні задачі комбінаторної оптимізації, для розв'язування яких у роботі пропонуються оригінальні метаевристичні алгоритми.

З практичної точки зору, при плануванні місій команди БПЛА три основні виклики включають розподіл цілей між базами, оптимізацію маршрутів і визначення оптимальних місць базування. Ці виклики часто розглядаються окремо в літературних джерелах [17, 18]. Проте в цьому дослідженні пропонується методологія, яка інтегрує всі ці проблеми в єдину задачу комбінаторної оптимізації.

Одним з ключових моментів проблеми є труднощі у прямому керуванні БПЛА у реальному часі через можливі радіоелектронні перешкоди. Це обумовлює те, що місія має бути докладно спланована наперед, а дрон повинен мати вбудовану програмну систему для автономного польоту. Комплексність ситуації, зумовлена численними місцями старту і посадки, робить ручне планування менш ефективним та точним. Отже, виникає потреба у розробці та застосуванні спеціальних програмно-алгоритмічних засобів планування місій БПЛА, використання яких буде спрямоване на мінімізацію часу виконання місії та зменшення затрат.

Варто також підкреслити, що такий планувальник може бути адаптований для використання у ряді інших завдань без великих змін. Це може включати завдання зі збору даних з датчиків для контролю за станом аграрних територій або координації роботи комбайнів при зборі врожаю. Багато завдань мають подібну структуру, тому можуть бути опрацьовані цим планувальником. Огляд наукової літератури та аналіз існуючих досліджень у даній сфері підтверджують актуальність розробки спеціальної системи планування.

Значний внесок в розвиток теорії і методів комбінаторної оптимізації, захисту інформації та управління в Україні зробили такі науковці як В.С. Михалевич, І.В. Сергієнко, В.П. Боюн, В.К. Задірака, І.М. Коваленко, А.О. Чикрій, Ю.Г. Стоян, П.І. Бідюк, Л.Ф. Гуляницький, О.О. Ємець, М.Ф. Каспшицька, М.Ю. Кузнєцов, О.А. Павлов, В.О. Перепелиця, М.М. Савчук, В.О. Трубін, С.В. Яковлєв та ін.

Мета й завдання дослідження. Метою роботи є розроблення і застосування математичних та програмно-алгоритмічних засобів, зокрема, на

основі ідей ройового інтелекту в плануванні місій окремих БПЛА (чи інших рухомих роботизованих систем – РРС) та їх команд при обстеженні чи обслуговуванні заданих цілей.

Основні *завдання* цього дослідження включають:

- розробити нові математичні моделі виділених проблем планування місій БПЛА;

- розробити прикладні алгоритми розв'язування задач комбінаторної оптимізації, що виникають, враховуючи специфічні обмеження у цій сфері;

- розробити програмні реалізації розроблених алгоритмів;

- провести дослідження ефективності розроблених алгоритмів на основі аналізу результатів обчислювального експерименту;

- здійснити реалізацію запропонованих програмно-алгоритмічних засобів у вигляді спеціального прикладного програмного забезпечення планування для місій.

Об’єкт дослідження – планування місій гібридних систем «Дрон + ТЗ» та команд гетерогенних безпілотних літальних апаратів.

Предмет дослідження – математичні моделі та методи оптимізації рішень при плануванні місій гібридних систем «Дрон + ТЗ» та команд гетерогенних безпілотних літальних апаратів..

Зв’язок роботи з науковими програмами, планами, темами. Дисертаційна робота виконана в рамках науково-дослідних робіт ВФ.180.14 "Розробити нові математичні моделі і методи комбінаторної оптимізації з використанням ройового інтелекту і нечітких множин, орієнтовані на оптимізацію рішень та захист даних у системах різного призначення", державний реєстраційний номер роботи 0122U000831, виконується за Постановою бюро Відділення інформатики НАН України від 08.07.2021 р. № 1 та ІК-2022/1”Розроблення та удосконалення математичних і апаратних засобів планування місій команд БПЛА та створення завадозахищених і шифрованих каналів зв’язку в системах управління безпілотними авіаційними комплексами та спеціальних інформаційно-телекомунікаційних мережах”, державний

реєстраційний номер роботи 0122U200274, виконується за розпорядженням Президії НАН України від 21.03.2022р. №151.

Методи дослідження. Виділені задачі планування місій розв'язуються запропонованими алгоритмами комбінаторної оптимізації, ефективність яких оцінюється на основі аналізу результатів чисельних експериментів.

Наукова новизна одержаних результатів. Наукову новизну в цій роботі мають такі теоретичні та практичні результати:

- вперше проведено аналіз новітніх досліджень (2020-2023 рр.) з питань оптимізації маршрутів окремих БПЛА та їх груп;
- вперше запропоновано математичну модель проблеми мінімізації часу виконання місії гібридного комплексу "Дрон+ТЗ", у якій враховується можливість незадіяності певних баз на маршруті транспортного засобу;
- вперше запропоновано математичну модель оптимізаційної проблеми планування місії команди БПЛА за наявності альтернативних баз (депо);
- вперше запропоновано ройовий алгоритм ОМК для розв’язування задач формування оптимального розкладу місії гібридного комплексу "Дрон+ТЗ" з мінімізацією часу її виконання;
- вперше запропоновано оригінальний метаевристичний алгоритм ройового інтелекту, який базується на ОМК з включенням запропонованих алгоритмів жадібного та локального пошуку, для розв’язування задач планування місії команди БПЛА за наявності альтернативних баз;
- вперше запропоновані алгоритми реалізовані у вигляді спеціального програмного комплексу, який інтегровано із поширеними системами керування БПЛА;
- вперше досліджено ефективність запропонованих ройових алгоритмів шляхом аналізу результатів обчислювальних експериментів;

- набула подальшого розвитку методологія розв’язування трудомістких задач комбінаторної оптимізації, базована на ройовому інтелекті.

Практичне значення отриманих результатів. Запропоновані в дисертації математичні моделі та методи оптимізації рішень при плануванні використання БПЛА та їх груп застосовуються для розв’язування задач комбінаторної оптимізації, що полягають у створенні плану місій гібридних систем та команд БПЛА. Створення таких місій є дуже важливим та затребуваним у таких галузях, як логістика та розвідка. Розроблений обчислювальний блок дозволяє формувати плани місій та передавати їх до розповсюджених систем керування, що здійснюють безпосередню взаємодію із БПЛА та гібридними системами. Розроблені алгоритми дозволяють планувати місії на персональних комп’ютерах чи серверах за обмежений час, що дозволяє створювати план без підключення до мережі Інтернет та оперативно коригувати його у разі необхідності. Проведений чисельний експеримент дозволив виявити більш ефективні алгоритми та методи для подальшої реалізації обчислювального блоку.

Значна частина результатів дисертації може бути використана у навчальному процесі вищих навчальних закладів України, де викладаються такі дисципліни як "Теорія прийняття рішень", "Комбінаторна оптимізація" та "Дослідження операцій".

Особистий внесок здобувача. Автором самостійно отримано основні результати дисертаційного дослідження або за участю його керівника. В опублікованих в співавторстві наукових працях здобувачем здійснено: у статті [19] розроблена математична модель задачі планування місії системи «Дрон + ТЗ», розроблено та реалізовано три алгоритми розв’язування задачі, досліджено їх ефективність; у статті [20] формалізовано проблему оптимізації місць базування та маршрутів групи БПЛА; у статті [21] формалізовано задачу планування місії команди БПЛА за наявності альтернативних баз та обмеженого ресурсу; у статті [22] розширено та узагальнено постановку задачі

планування місії команди гетерогенних БПЛА, розроблено її математичну модель, розроблено три алгоритми її розв'язування, досліджено їх ефективність. Співавтору статей належить постановка задачі, ідея і методологія проведення досліджень.

Апробація результатів дисертації. Результати дисертації доповідались та обговорювались на:

• III International Scientific Symposium «Intelligent Solutions-2023», September 27-28, 2023, Kyiv, Ukraine;

• засіданнях наукового семінару "Методи комбінаторної оптимізації та інтелектуальні інформаційні технології" Інституту кібернетики імені В.М. Глушкова НАН України впродовж навчання в аспірантурі;

• засіданні розширеного семінару "Методи комбінаторної оптимізації та інтелектуальні інформаційні технології" Інституту кібернетики імені В.М. Глушкова НАН України 26 жовтня 2023 р.

Публікації. Основні наукові результати дисертаційної роботи у повній мірі викладено в 4 публікаціях, з яких: 3 статті опубліковано в фахових виданнях України, 1 з них опубліковано англійською мовою та проіндексовано в наукометричній базі SCOPUS у збірнику доповідей III International Scientific Symposium "Intelligent Solutions-2023", September 27-28, 2023, Kyiv, Ukraine .

Структура та обсяг дисертації. Дисертаційна робота складається зі вступу, семи розділів, загальних висновків, списку використаних літературних джерел, який містить 67 найменувань. Загальний обсяг дисертаційних досліджень викладено на 107 сторінках друкованого тексту, де обсяг основного тексту – 87 сторінок. Дисертація включає 26 рисунків та 1 додаток на 4 сторінках.

РОЗДІЛ 1. ОГЛЯД РЕЛЕВАНТНИХ ДОСЛІДЖЕНЬ

1.1 Про задачі планування місій з використанням БПЛА

Можна виокремити три основні способи використання БПЛА в логістиці, моніторингу чи розвідці:

– обслуговування здійснює лише БПЛА;

– для обслуговування задіюють як транспортний засіб, так і один чи декілька БПЛА;

– транспортний засіб обслуговує лише БПЛА.

Як уже зазначалося, наявні певні аналогії із VRP з декількома пунктами базування (депо, базами) – стаціонарними чи динамічними [8–12].

У науковій літературі існує ряд термінів та скорочень для опису задач маршрутизації з використанням БПЛА: UAVRP [23], EVRP [3], FSTSP [25], VRPD [2, 26, 27, 28] та VRP-T [29, 12]. Ці задачі розглядаються як розвиток і адаптація традиційних задач VRP, проте вони володіють власною унікальністю [23, 12, 30, 31, 32].

У сфері логістики, розподіл товарів є ключовою операційною діяльністю і охоплює транспортування від виробника до кінцевого споживача транспортною системою. Організація такого процесу потребує великих капіталовкладень, особливо у сфері діяльності служб доставки. Термін VRP позначає клас задач комбінаторної оптимізації, де клієнти обслуговуються певною кількістю транспортних засобів. Транспортні засоби базуються у вантажних відділеннях, виходять для здійснення обслуговування та потім повертаються. Ця задача має численні варіанти постановки обмежень, таких як час доставки, вантажопідйомність, максимальна довжина маршруту та ін. [33].

Специфіка прикладних задач з використанням БПЛА, які розглядаються, вносить корективи до тлумачення як терміну "вартість перевезень", так і "транспортний засіб". Особливий клас задач породжують ситуації, у яких БПЛА можуть стартувати із декількох, зокрема рухомих пунктів базування та фінішувати в низці цих чи інших пунктів – такі задачі актуальні, перш за все, у

логістиці і військовій сфері. При оптимізації маршрутів у цьому випадку вантажопідйомність БПЛА може не грати ключову роль (наприклад, при аеророзвідці), але більш суттєве значення мають обмеження на ресурс дрона – обсяг пального чи ємність акумулятора.

У публікаціях, що стосуються проблем маршрутизації транспортних засобів з фіксованими місцями розміщення вживається термін депо, але з появою задач, де у якості місць можливого обслуговування виступають стаціонарні чи мобільні станції або місця на маршруті, більш доречним вважається використання терміну база. Тож у дослідженні вживається саме цей термін.

Суттєвою причиною активізації останнім часом досліджень і розроблень із цієї тематики, зокрема, є те, що низка великих компаній, таких як Amazon, Google, DHL, UPS, Deutsche Post AG та Walmart оголосили про використання безпілотників у процесі доставлення товарів кінцевим споживачам ("доставка останньої милі") [34]. Розширюються дослідження задач мінімізації загального споживання енергії при плануванні логістичних місій з використанням дронів (Electric Vehicle Routing Problem with Drones), де транспортний засіб – це електричні фургони, з яких БПЛА здійснюють доставлення останньої милі [35].

Важливими є також задачі планування моніторингу об'єктів критичної інфраструктури групою БПЛА з урахуванням вимог, таких як частота передачі, обсяг переданих даних та час безперервної роботи, планування використання БПЛА та станцій автоматичної заміни акумуляторів [36]. Також зазначимо, що задачі маршрутизації у багатьох випадках пов'язані з моніторингом, зокрема для оперативного виявлення заданих об'єктів чи процесів на певній території, що з математичної точки зору приводить до задач максимального покриття області сукупністю геометричних об’єктів заданої форми та розмірів [37].

В [23], одній із ранніх робіт, увага акцентується на основних принципах дії планувальника місії (pre-mission planner) за наявності визначених цілей, зосереджуючи увагу на оптимізації витрат та мінімізації ризику потрапляння у зону дії потенційних загроз, серед яких вплив атмосферних умов, зокрема

вітру. Детально розглядаються як детерміновані, так і стохастичні методи формування плану місії.

Особливістю планування багатьох місій з БПЛА є використання гібридних транспортних систем, у яких БПЛА може транспортуватися автомобілем чи іншим транспортним засобом, злітаючи і приземляючись у певних місцях на маршруті транспортного засобу [33, 38 – 41]. Окремий клас таких задач складають проблеми планування місій гібридних систем, у яких місця базування транспортного засобу можуть задаватися без прив'язки до об'єктів обслуговування. При цьому місцеположення таких баз можуть задаватися апріорі (подібно до задач із динамічними базами [42]) чи визначатися у процесі розв'язування задачі. В [43] пропонується такі задачі називати Drone Routing Problem with Truck (DRP-T). У теоретичному плані формальні постановки зазначених проблем приводять до задач комбінаторної оптимізації підвищеної складності.

Найновіші дослідження стосуються питань оптимізації рішень при плануванні місії гібридних систем, які складаються із транспортного засобу, який у ході виконання завдань може перевозити один чи декілька БПЛА. Більшість публікацій розглядає як транспортний засіб саме вантажівку, але у подальшому ми не будемо обмежуватися лише вантажівками. Такі проблеми із застосуванням гібридних транспортних систем у логістиці зазвичай характеризуються поєднанням "Дрон+Вантажівка" ("Drone+Truck"), вони породжують цілий клас задач оптимізації. Автори досліджень таких задач враховують різні характеристики, умови та обмеження, що виникають на практиці, чим і пояснюється різноманіття назв конкретних задач. Як і в більшості публікацій, далі будемо теж вживати назву дрон, розуміючи саме БПЛА.

У першій публікації, де розглядалась взаємодія одного дрона з вантажівкою при доставці товарів, припускалося, що за один виліт дрон обслуговує одного клієнта, а відповідна проблема оптимізації була названа Flying Sidekick Traveling Salesman Problem (FSTSP) – задача комівояжера з

літаючим помічником [44]. В роботі [45] розглядається задача, у якій дрон за один виліт може обслуговувати уже декількох клієнтів, місця зустрічі вантажівки та дрона не визначено заздалегідь, а вантажівка теж бере участь в обслуговуванні клієнтів. Робота [46] зосереджується на питаннях економії ресурсів як дрона, так і вантажівки. В [47] пропонується гнучка методологія підзарядки акумулятора дрона, яка відкидає припущення, що акумуляторну батарею слід замінювати повністю зарядженою після кожного польоту. Автори роботи [47] припускають, що дрон може злітати з вантажівки, а приземлятися для поповнення ресурсу у одному із сервісних центрів, щоб продовжити рух, можливо, з іншої вантажівки. В [48] розглядається варіант задачі мінімізації загальних витрат системи "дрон+вантажівка", у якій дрон може не тільки обслуговувати декількох користувачів за один політ, а і поєднувати доставку із збиранням посилок від користувачів. При цьому при розрахунку затрат польотного ресурсу враховується не лише довжина маршрутів, а і вага вантажів.

Розширенню класу задач з одним транспортним засобом та дроном присвячена робота [49], де материнський корабель може нести один дрон – менший транспортний засіб, який перевозиться кораблем з можливістю запуску або повернення дрона в будь-якому місці, а не лише в певних вузлах графа. Дрон запускається до певного місця, а потім повертається на корабель для дозаправки або для того, щоб забрати новий вантаж перед наступним запуском.

Наступним кроком стали дослідження питань обслуговування групою дронів, зокрема при доставці "останньої милі", які транспортуються однією вантажівкою. Така задача в [50] названа multiple flying sidekicks traveling salesman problem (mFSTSP), де, на відміну від [44], дронам дозволено відвідувати декількох споживачів за один політ. Розширена версія подібної задачі в [51] названа Truck-multi-Drone Team Logistics Problem (TmDTL), яка полягає в тому, щоб за мінімальний час, починаючи з певного місця і закінчуючи фінальним місцем на маршруті вантажівки, відвідати вантажівкою

разом з командою дронів задану множину клієнтів, доставляючи легкі вантажі чи виконуючи місію розвідки, спостереження і рекогносцировки.

Задача маршрутизації вантажівки з групою дронів, які можуть доставляти по декілька пакетів одному чи більше користувачам, здійснюючи по декілька польотів, розглядається в [52], де вона названа *k*-Multi-visit Drone Routing Problem (*k*-MVDRP). Вантажівка виконує лише роль підтримки дронів (поповнення ресурсу, завантаження посилками). Тут, як і в ряді нових робіт, при розрахунку витрат енергоресурсу дронів враховується і вага пакетів, які доставляються [53, 54].

В [55] для зменшення загального часу місії допускається повторне відвідування вантажівками місць обслуговування. Мінімізація експлуатаційних затрат в [56] досягається шляхом вибору швидкості дрона, значення якої вибирається із дискретного набору. В [57] також пропонується швидкість дрона у задачі TSP-D визначати динамічно відповідно до ваги вантажу, який має перевозити дрон; при цьому один дрон за один політ може здійснити доставку лише одному користувачу. Задачі, у яких місця запуску дронів можуть знаходитися окремо від розташування клієнтів, стали об'єктом досліджень в [58], а їх розвиток, відображений у [59], стосується аналізу та розв'язування проблем створення місць для базування вантажівки, у яких забезпечується поповнення ресурсу дронів та їх завантаження.

До цього ж типу задач можна віднести і задачі оптимізації маршрутів команди гетерогенних дронів, які запускаються і приймаються великим транспортним літаком у виділених місцях траєкторії його польоту [60].

Враховуючи важливість тематики, з'явилося і з'являється велика кількість наукових публікацій, тому, в основному, виділено нові публікації, які відносяться до періоду 2020-2023 рр. і не є вичерпним оглядом релевантних робіт; більше інформації, особливо стосовно попереднього періоду досліджень, можна почерпнути із оглядових статей – див., напр., [33, 53, 60- 63].

Переваги застосування гібридних систем, які далі будемо називати "Дрон+ТЗ" ("Drone+Vehicle"), розуміючи під ТЗ деякий транспортний засіб, а

не лише вантажівку, мають свою ціну. Для ефективного використання таких систем обслуговування необхідна розробка і застосування як адекватних математичних моделей виникаючих проблем, так і ефективних алгоритмів оптимізації рішень при плануванні та прогнозування результатів їх місій.

1.2 Огляд та класифікація основних класів наближених методів комбінаторної оптимізації

Оскільки на задачах великої розмірності застосування точних алгоритмів є практично недоцільним через експоненційно зростаючу кількість обчислень, у більшості випадків застосовуються наближені алгоритми комбінаторної оптимізації (АКО). Необхідність розробки ефективних наближених АКО, які застосовуються у переважній більшості випадків на практиці, визначається низкою обставин:

- переважна більшість важливих задач належать до NP-складних, тож точне їх розв'язання дуже проблематичне навіть із використанням сучасних і перспективних комп'ютерів;
- їх цільові функції зазвичай мають велику кількість локальних екстремумів;
- у багатьох прикладних проблемах дані задаються з певними похибками, що робить недоцільними ті істотні обчислювальні затрати, які необхідні для знаходження їх точного розв'язку;
- покладені в основу розробки наближених обчислювальних схем ідеї (метаевристики) дозволяють створювати алгоритми, які можуть розв'язувати не одну, а цілий клас близьких за постановкою оптимізаційних задач;
- важливий клас оптимізаційних задач породжується проблемами з директивним терміном, тобто їх розв'язок має бути знайдений до зазначеного апріорі строку;

- у деяких задачах значення цільової функції можуть бути доступними лише в процесі розв'язання задачі або змінюватися з часом – цей клас утворюють динамічні, або on-line задачі.

Для дослідження складності алгоритмів задач КО такі задачі розглядаються як задачі розпізнавання вигляду: для даної ЗКО і цілого числа M визначити, чи існує такий припустимий розв'язок $x \in D \subseteq X$, що $f(x) \le M$. Відповідно, для задачі максимізації умова мала би вигляд $f(x) \ge M$.

Задачі класу P можуть бути розв’язані алгоритмами з поліноміальною складністю. Ці алгоритми мають обмеження на складність, яка виражена поліномом, залежним від вхідних даних. Таким чином, задачі цього класу вважаються відносно простими та мають ефективні рішення. Задачі класу NP розв'язуються недетермінованими поліноміальними алгоритмами, що складаються з двох етапів: знаходження потенційного розв'язку та його перевірка. Якщо задача відповідає певним критеріям, її можна визнати NP-повною. Якщо задача відносно складна, але не підтверджена як NP-повна, її класифікують як NP-складну. Важливим питанням є рівність P і NP. Якщо вони не рівні, немає гарантії поліноміального розв'язку для задач NP, тому акцент робиться на розробці ефективних наближених алгоритмів.

Наближені алгоритми хоча і не завжди забезпечують знаходження точного розв'язку, можуть дати дуже хороший результат за відносно короткий час. Велике значення для практики має здатність цих алгоритмів ефективно опрацьовувати великі набори даних і знаходити оптимальні або майже оптимальні рішення.

Важливо розуміти, що незважаючи на те, що питання про рівність відомих класів складності задач P і NP залишається відкритим, воно має глибокий теоретичний і практичний зміст. Якщо колись буде доведено, що P не дорівнює NP, це може змінити підхід до розробки алгоритмів і стратегій розв'язання задач. До того ж, вивчення NP-повних задач дає можливість

розробляти нові математичні та комп'ютерні методики, які можуть бути застосовані в інших дисциплінах.

Вважатимемо для простоти, що розглядається ЗКО без обмежень: необхідно знайти такий елемент $x_* \in X$, що:

$$x_* = \arg \underset{x \in X}{ext} f(x). \tag{1.1}$$

Вважатимемо, що алгоритм комбінаторної оптимізації (АКО) – це певна процедура A, що переводить задану підмножину $Z \subset X$ (початкові наближення) у множину $X_* \subset X$:

$$AZ = X_* \tag{1.2}$$

тобто X_* – множина розв’язків задачі.

Існують різні підходи до класифікації АКО – за структурою обчислювальної схеми, точністю, типом використаних просторів тощо (рисунок 1.1).

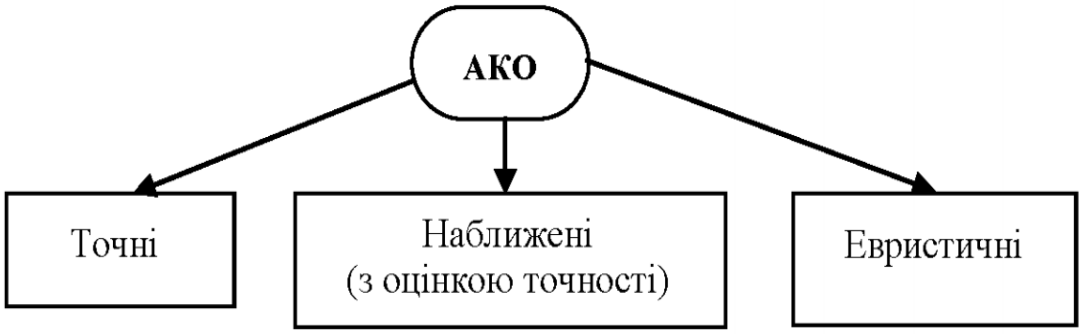

Рисунок 1.1. Загальна класифікація АКО за точністю [2].

Точні АКО – це методи, для яких $X_* \subseteq \arg ext$ (знаходять глобальний розв’язок). Наближені АКО поділяють на алгоритми з апріорною та апостеріорною оцінками точності. Евристичні алгоритми базуються на правдоподібних міркуваннях (наприклад, алгоритм "йди в найближче місто"

для задачі комівояжера), проте не можуть оцінити точність знайденого розв’язку. Часто дослідники називають наближеними як алгоритми з оцінкою точності, так і евристичні.

За типом обчислювальної схеми наближені АКО поділяють на конструктивні та ітераційні. Конструктивні алгоритми послідовно формують повний розв’язок, починаючи "з нуля" чи фрагмента розв'язку. Ітераційні алгоритми на кожному кроці опрацьовують "повні" розв’язки, намагаючись поліпшувати поточний розв’язок. За складністю структури АКО поділяють на: прості; комбіновані; метаевристики; гібридні метаевристики; гіперевристики. За впливом на ландшафт пошуку АКО поділяють на такі, що залишають його незмінним, та такі, що модифікують цей ландшафт. За типом використання даних АКО діляться на задаче-орієнтовані та моделе-орієнтовані.

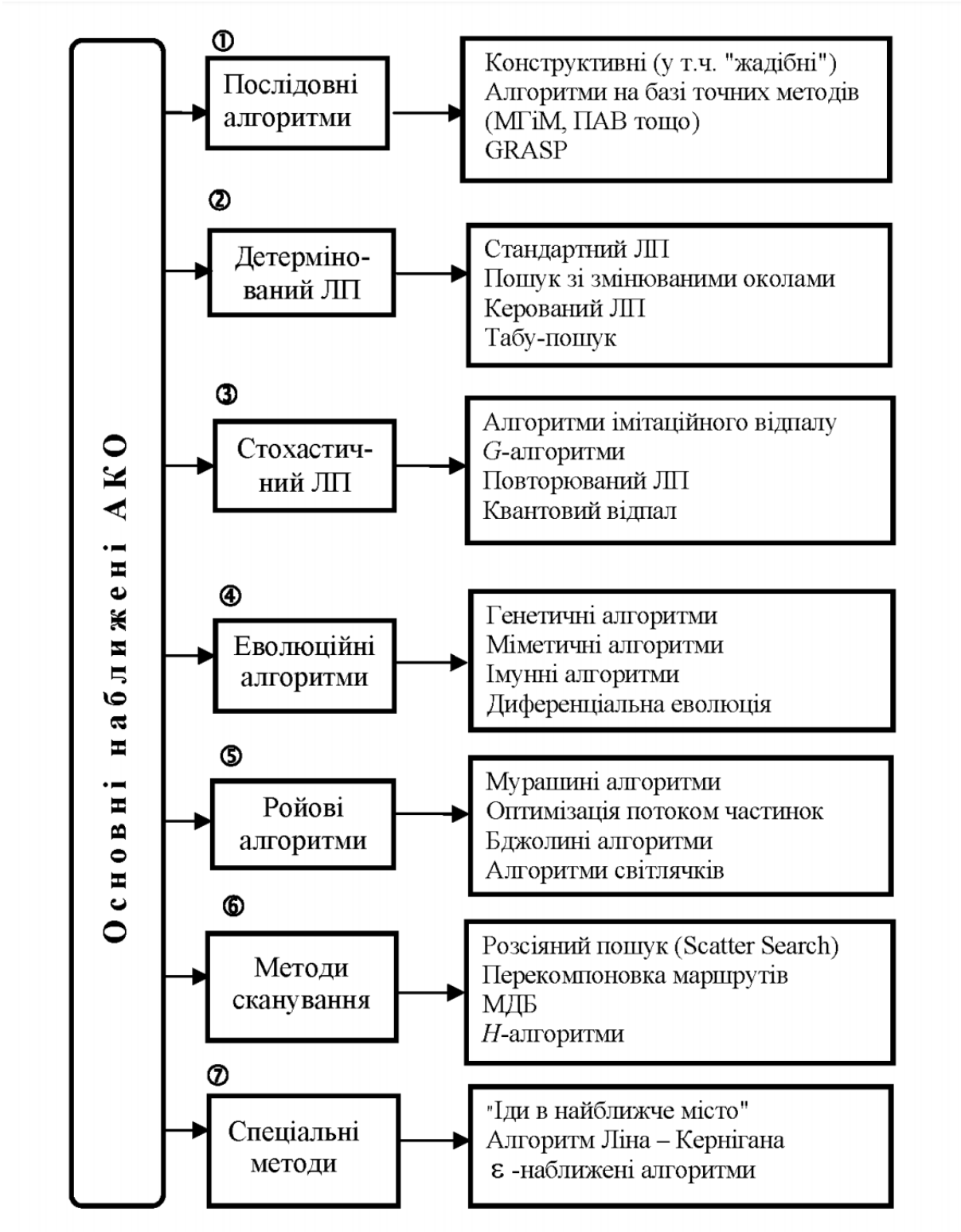

Рисунок 1.2. Класифікація основних наближених методів комбінаторної оптимізації [64].

Найуживаніші на практиці наближені алгоритми можна поділити на сім класів (рисунок 1.2) [64]. Тут МГіМ – це метод гілок і меж, ПАВ – послідовний

аналіз варіантів; ці точні алгоритми використовуються в даному контексті для породження наближених обчислювальних схем. Зрозуміло, що сукупність усіх розроблених донині АКО значно перевищує перелік наведених на цьому рисунку. Проте, як свідчить аналіз наукових публікацій, саме зазначені алгоритми та їх модифікації є основним інструментом для розв'язання практичних ЗКО.

Левову частку алгоритмів із класів 2-7 становлять ітераційні методи, багато з яких базуються на використанні процедур локального пошуку. Крім того, для розв'язання задач із підвищеною точністю значного поширення набули метаевристики.

Метаевристики класифікують відповідно до парадигми, яка використовується при їх побудові. У [65] наведено класифікацію та приклади поширених метаевристик:

- Детермінований локальний пошук - це метаевристики ітераційного типу, в основі яких лежить частковий перебір варіантів на кожній ітерації в околі поточного розв'язку. Поширені метаевристики цього типу: пошук зі змінюваними (пульсуючими) околами; керований локальний пошук; табу-пошук.
- Метаевристики на основі стохастичного локального пошуку використовують у якості підлеглої процедуру локального пошуку, у яку вбудовано ймовірнісні механізми. Приклади: метод звужувальних околів; алгоритми імітаційного відпалу; О-алгоритми; повторюваний локальний пошук; квантовий відпал; метод кросс-ентропії; GRASP.
- Еволюційні метаевристики - це алгоритми пошуку, що використовуються для розв'язання задач оптимізації і моделювання шляхом послідовного відбору, комбінування і варіації шуканих параметрів розв'язків із використанням механізмів, що нагадують біологічну еволюцію. Приклади: генетичні алгоритми; міметичні алгоритми; імунні алгоритми.
-

- Метаевристики на основі ройового інтелекту – децентралізовані системи простих агентів, які локально взаємодіють як з середовищем, так і між собою. Приклади: мурашині алгоритми; оптимізація роєм часток; бджолині алгоритми; алгоритми світлячків; алгоритми моделювання поведінки бактерій.
- Методи сканування простору - популяційні метаевристики, на кожній ітерації яких формуються напрями нелокального пошуку в просторі розв'язків на основі наявних декількох варіантів наближень. Приклади: розсіяний пошук; перекомпоновка маршрутів; *H*-метод.

Окремо розглянемо алгоритми оптимізації мурашиною колонією (ОМК; ant colony optimization – ACO), що широко застосовуються для розв’язування задач маршрутизації різного типу транспорту [66].

Мурашині алгоритми – це багатоагентні системи, де поведінка кожного агента, який називається штучною мурахою, або просто мурахою, заснована на поведінці справжніх мурашок. Вони успішно застосовуються для розв’язування багатьох типів задач комбінаторної оптимізації.

В алгоритмах ОМК формується спеціальна модель задачі, що розв'язується, тому вони належать до класу моделе-орієнтованих методів. Така модель задачі подається у вигляді повного зваженого графа $G = (V, E)$, де V – вершини, що відповідають компонентам розв'язку, а E – ребра, які відповідають можливим з'єднанням (переходам) між відповідними вершинами.

Розв'язки задачі оптимізації можуть бути подані як припустимі шляхи на графі G. Алгоритми ОМК можуть використовуватися для знаходження припустимих шляхів мінімальної вартості, що задовольняють обмеженням задачі.

У мурашиних алгоритмах популяція агентів (або мурашок) спільно розв’язує сформульовану задачу оптимізації, використовуючи вищезазначене подання на графі моделі задачі [67]. Мурахою в таких алгоритмах фактично є параметричний рандомізований жадібний алгоритм, який покроково будує з

множини компонент (вершин чи ребер графа задачі) припустимий розв'язок задачі.

Отже, огляд показує, що метаевристичні алгоритми, зокрема мурашині, є ефективним підходом для розв'язання складних задач комбінаторної оптимізації, якими є задачі маршрутизації. Тому їх використання є доцільним і для оптимізації маршрутів групи БПЛА.

1.3 Висновки до розділу

Задачі планування місій з використанням безпілотних літальних апаратів набувають все більшої актуальності у зв'язку з поширенням їх застосування в логістиці, моніторингу та інших сферах. Існує велика різноманітність постановок таких задач залежно від кількості БПЛА, наявності інших транспортних засобів, обмежень на ресурси тощо.

Оскільки ці задачі належать до класу NP-складних задач комбінаторної оптимізації, застосування точних методів для їх розв'язування є практично неможливим. Тому широко використовуються різноманітні наближені методи, серед яких значне місце займають метаевристичні підходи, зокрема методи оптимізації мурашиною колонією. Вони дозволяють ефективно знаходити наближені розв'язки задач великої розмірності та можуть бути застосовані і для оптимізації маршрутів груп БПЛА.

Аналіз літературних джерел продемонстрував, що розробка і використання сучасних ефективних метаевристичних алгоритмів для задачі планування місії гібридної системи та задачі планування місії команди гетерогенних БПЛА є важливими та актуальними напрямками досліджень.

РОЗДІЛ 2. ЗМІСТОВНІ ПОСТАНОВКИ ЗАДАЧ

2.1 Вступні положення

Планування місій для гібридних систем та груп БПЛА є складним завданням, що потребує врахування багатьох аспектів. У цьому розділі детально описано два ключові класи задач, які лягли в основу дослідження.

Перша задача стосується планування маршрутів для одного дрона, що переміщується разом із транспортним засобом. Сформульовано початкові умови та припущення: фіксовані місця базування дрона, обмеження на його ресурс, можливість поповнення ресурсу на базах. Визначаються основні компоненти задачі: транспорт, дрон, цілі для відвідування, бази. Метою є оптимізація маршрутів дрона з урахуванням заданих умов та обмежень.

Друга задача полягає в оптимізації маршрутів для групи гетерогенних БПЛА, що мають виконати задану місію. Означаються початкові умови: характеристики БПЛА, цілі, бази, обмеження. Метою є мінімізація загальних витрат ресурсів усіма залученими дронами.

Також описуються основні етапи розв'язування задач. Обидві задачі характеризуються високою складністю через велику кількість змінних, цілей та обмежень. Їх успішне розв'язування потребує розробки відповідних моделей та ефективних алгоритмів оптимізації. Представлені постановки задач становлять концептуальну основу для подальшого дослідження.

2.2 Про основні проблеми планування місій з гібридними системами

Розглядаються проблеми планування місії гібридної системи "Дрон + ТЗ", які полягають у виборі місць базування, розподілу цілей і оптимізації маршрутів одного дрона при виконанні поставленого завдання. Особливістю розглянутих далі проблем є те, що дрон переміщується спеціальним транспортом і при цьому мають виконуватися умови початку і завершення маршруту в місцях на траєкторії руху транспорту за наявності обмежень на ресурс дрона, але у деяких місцях запуск дрона може і не відбуватися.

В подальшому під транспортом розуміється засіб (автомобіль, літак, корабель тощо), який може перевозити (переміщувати) безпілотні апарати (дрони, надводні чи підводні роботизовані системи). Місця на маршруті транспорту, де може відбуватися старт, приймання і обслуговування дронів, називатимемо базами.

Серед задач маршрутизації дронів за наявності декількох баз виділимо такі, в яких:

1) місця знаходження баз визначені на заданій траєкторії руху транспорту;

2) ці місця формуються в процесі планування місії гібридної системи;

3) бази можуть використовуватися для відновлення енергоресурсів дрона (підзаряджання акумуляторів, заправка паливом) з подальшим продовженням процесу виконання завдань;

4) метою є відвідання максимальної кількості цілей (можливо, з урахуванням їх важливості), якщо наявний склад дронів та їх ресурси не дозволяють обслужити всі цілі.

У подальшому особлива увага буде приділена першому та третьому із названих типів задач. Ці дослідження є розвитком поданих в [14, 20] результатів.

При плануванні місій гібридної транспортної системи, пов'язаних із обстеженням чи обслуговуванням дронами заданої сукупності цілей (користувачів), можна виділити такі етапи:

1) Формування сукупності цілей для обстеження чи обслуговування.

2) Визначення місць розташування баз на маршруті транспорту.

3) Вибір баз для обслуговування дрону.

4) Розподіл цілей між всіма базами чи деякими із них.

5) Формування маршрутів дрону із урахуванням обмежень на польотний ресурс.

6) Підтримка функціонування мобільної мережі, що зв'язує дрон та транспорт, захист таких мереж та оптимізація передачі інформації в них.

Ці етапи породжують ряд задач, серед яких домінують задачі оптимізації, зокрема, комбінаторного типу. В низці публікацій такі задачі розглядаються як окремі, хоча на практиці часто слід враховувати їх взаємозв'язок.

Варто зазначити, що у випадку нестачі ресурсу дрону виникають задачі охоплення такої кількості цілей, які б максимізували ефект від застосування гібридної системи із урахуванням наявних характеристик цілей та дрону, а також обмежуючих умов.

Предметом подальшого розгляду стали задачі етапів 3-5, тобто вважається, що цілі визначено, а ресурсу дрону достатньо для їх обслуговування. Зрозуміло, що завдання носять оперативний характер, тобто як перелік цілей, так і маршрут транспорту може змінюватися щоденно чи навіть з меншим часовим інтервалом.

Виділимо два можливих напрями агрегування проблем: на рівні математичних моделей виникаючих задач (наприклад, задач розбиття множини цілей та задач комівояжера, які охоплюють етапи 4 та 5) чи/та шляхом розроблення алгоритмів, які орієнтовані на одночасне розв'язування задач, що виникають на кількох етапах.

У цій роботі розглянуто одну із центральних проблем планування місії гібридної системи "Дрон + ТЗ" – розподіл цілей по базам, вибір задіяних баз із можливих та оптимізацію маршрутів дрона – перед якою стоїть завдання обстежити чи обслужити задану множину цілей на місцевості.

2.3 Змістовні постановки досліджуваних проблем оптимізації місій гібридної системи з одним дроном

Задачі планування місії гібридної системи "Дрон + ТЗ" за наявності декількох баз на маршруті транспорту розглядається з урахуванням таких припущень.

1) Кожна ціль відвідується дроном один і тільки один раз. Транспорт в обстеженні цілей участі не бере.

2) Місцезнаходження баз на заданому маршруті транспорту вважаються фіксованими, а транспорт може відвідувати їх послідовно без повернень, починаючи із першої бази, з якої і може стартувати дрон.

3) За один політ дрон може відвідати декілька цілей (мультивізитність).

4) Дрон має обмеження на польотний ресурс.

5) Поповнення ресурсу дрона здійснюється на базах (зупинках транспорту).

6) Запасів для поповнення ресурсу дрона достатньо (акумулятори, пальне) для виконання місії.

7) Споживання енергії дроном відбувається за лінійним законом, тобто не враховуються можливі перевищення під час зльоту чи приземлення дрона чи під дією вітру.

8) Переліт між цілями чи цілями і базами здійснюється по прямій лінії, тобто не враховуються кути поворотів.

9) Загальний маршрут дрона може складатися з підмаршрутів, кожен із яких починається і закінчується на заданій базі маршруту транспорту, а час обстеження включених у кожний підмаршрут цілей не порушує обмеження на польотний ресурс. При цьому вибір баз та цілей для включення у підмаршрут здійснюється у ході виконання місії.

10) Транспорт може прибувати на чергову базу раніше дрона і очікувати на нього.

11) Транспорт починає рух до наступної бази, якщо вона не остання на його маршруті, від поточної бази після повернення дрона із останнього із запланованих на цій базі підмаршрутів.

12) Швидкість дрона є вхідною умовою задачі та не змінюється протягом виконання місії.

Отже, маємо транспортний засіб, один дрон, задану множину цілей для обстеження та місцезнаходження послідовності баз на маршруті транспортного засобу. Дрон стартує з транспортного засобу, що знаходиться в даний момент на певній базі, починаючи із першої, а сам транспортний засіб після

проходження підмаршруту дроном залишається на місці, якщо не всі заплановані підмаршрути пройдено, чи вирушає до наступної бази, якщо він не знаходиться на кінцевій базі.

2.4 Змістовна постановка досліджуваної проблеми планування місій команди БПЛА

Пропонована постановка задачі планування місії групи гетерогенних БПЛА з мінімізацією затрат є розвитком постановок UAVRP, запропонованих в [20, 21].

Метою розв’язування UAVRP є знаходження оптимальних траєкторій для кожного БПЛА в команді таким чином, щоб максимізувати загальну ефективність місії, мінімізувати час виконання або витрати палива, а також забезпечити відповідність всім обмеженням.

Сформульована задача розв'язується за таких припущень.

1) Кожну ціль відвідує лише один БПЛА і тільки один раз.

2) БПЛА мають обмеження на польотний ресурс.

3) Поповнення польотного ресурсу БПЛА здійснюється в одній із наявних базах, яка визначається у ході планування.

4) Вважається, що засобів і запасів для поповнення польотного ресурсу БПЛА достатньо (акумулятори, пальне).

5) Вважається, що споживання енергії БПЛА відбувається за лінійним законом, тобто не враховуються можливі перевищення витрат під час зльоту чи приземлення БПЛА.

6) Маршрут конкретного БПЛА може складатися з підмаршрутів, кожен із яких починається і закінчується у заданих базах, причому конкретний БПЛА може стартувати з однієї, а повертатися на іншу базу.

7) Вибір стартових і фінішних баз та цілей для включення у підмаршрути здійснюється у ході виконання місії.

8) Розглядаються завдання для БПЛА, у яких їх вантажопідйомність не є обмежуючим чинником (моніторинг, обстеження, доставка легких предметів).

9) З міркувань доцільності деякі бази та БПЛА можуть бути незадіяними.

Можна виділити наступні основні компоненти поданої задачі планування місії команди БПЛА:

1) Набір агентів: група БПЛА, які беруть участь у місії. Кожен БПЛА має фіксовані характеристики: дальність польоту, швидкість.
2) Набір цілей, які мають бути обслуговані під час виконання місії. Кожна ціль має визначене місцезнаходження.
3) Набір баз, які можуть використовуватися для вильоту, прильоту та обслуговування БПЛА.
4) Обмеження: різні типи обмежень, які мають бути забезпечені під час планування місії. Це можуть бути часові обмеження (наприклад, часові вікна для виконання завдань), технічні обмеження (наприклад, максимальна дальність польоту БПЛА) та інші типи обмежень.
5) Визначення цілі місії, яка може включати мінімізацію часу місії, мінімізацію витрат ресурсів, максимізацію кількості виконаних завдань та інші цілі.

Загальні кроки для розв'язування поданої задачі можуть бути означені наступним чином:

1) Визначення початкового стану: встановлення місцезнаходження БПЛА або способу його визначення, їхнього стану, доступних ресурсів та інших параметрів.
2) Створення математичної моделі, що відображає всі аспекти місії, включаючи характеристики БПЛА, обмеження, граф задачі та цільову функцію.
3) Розробка алгоритмів оптимізації для знаходження допустимих маршрутів для БПЛА з урахуванням усіх обмежень та цільової функції.

4) Аналіз результатів: оцінка отриманих маршрутів з точки зору обраної цільової функції.
5) Інтеграція отриманих результатів з системами керування групами БПЛА для прикладного застосування створених планів місій.

Складність поданої задачі полягає у великій кількості змінних, обмежень та цілей, а також необхідності врахування динамічних змін у середовищі (наприклад, погодних умов, перешкод на маршруті). У роботі розглянуто виключно планування місії, тож моніторинг, динамічна зміна ситуації та безпосереднє керування БПЛА у дослідженні не розглядаються.

2.5 Висновки до розділу

Предметом розгляду стали змістовні постановки двох запропонованих задач планування місій з використанням гібридної системи та команди БПЛА, що є важливими елементами дослідження.

Перша змістовна постановка описує задачу оптимізації маршрутів одного дрона, якого супроводжує транспортний засіб. Сформульовано вихідні дані та припущення: фіксовані бази дрона, обмеження на ресурс, можливість поповнення ресурсу. Визначено такі ключові елементи: транспорт, дрон, цілі, бази. Метою є побудова оптимального маршруту дрона з урахуванням заданих умов та обмежень. Ця задача є складною комбінаторною оптимізаційною проблемою через велику кількість можливих варіантів маршрутів, задіяних баз та БПЛА.

Друга задача полягає в оптимізації маршрутів для команди гетерогенних БПЛА з метою виконання заданої місії. Визначено вихідні дані: характеристики дронів, цілі, бази, обмеження. Метою є мінімізація сумарного споживання ресурсів усіма задіяними дронами. Ця багатоцільова задача також є дуже складною через велику кількість змінних та параметрів.

Розглянуті змістовні постановки задач є важливою концептуальною основою для побудови відповідних математичних моделей. Успішне розв'язування цих задач вимагає ретельного аналізу, побудови адекватних

математичних моделей та розробки ефективних обчислювальних методів. Подальша робота спрямована на формалізацію задач, створення таких моделей і алгоритмів з урахуванням поданої специфіки.

РОЗДІЛ 3. МАТЕМАТИЧНІ МОДЕЛІ ЗАДАЧ

3.1 Вступні положення

У цьому розділі представлені математичні моделі для двох ключових задач дослідження – планування місії гібридної системи та планування місії команди гетерогенних БПЛА.

Для задачі планування місії гібридної системи побудована математична модель, що враховує основні компоненти задачі: дрон, транспорт, цілі, бази, обмеження на ресурс дрона. Для задачі планування місій команди БПЛА побудована математична модель, що враховує агентів (БПЛА) та їх характеристики, цілі, бази, обмеження ресурсів.

Розроблені моделі дозволили формалізувати задачі та стали основою для розробки алгоритмів їх розв’язування у наступних розділах.

Маршрутом БПЛА називається впорядкований набір вершин графу задачі, що відповідає наступним умовам:

- починається з бази, з якої дозволено вильоти;
- закінчується у базі, на яку можна здійснити посадку;
- містить вершини, що є цілями або базами;
- перельоти між початковою та кінцевою базою за довжиною не перевищують максимальну дальність польоту БПЛА.

У результаті планування місії кожен БПЛА отримує призначення у вигляді набору допустимих маршрутів, або помічається як незадіяний у місії.

Щоб визначити відстань між точками на Землі, використовується формула гаверсинуса, також відома як геодезична відстань. Це ключове рівняння у навігаційній сфері, яке дозволяє розраховувати відстань, базуючись на довготі та широті точок.

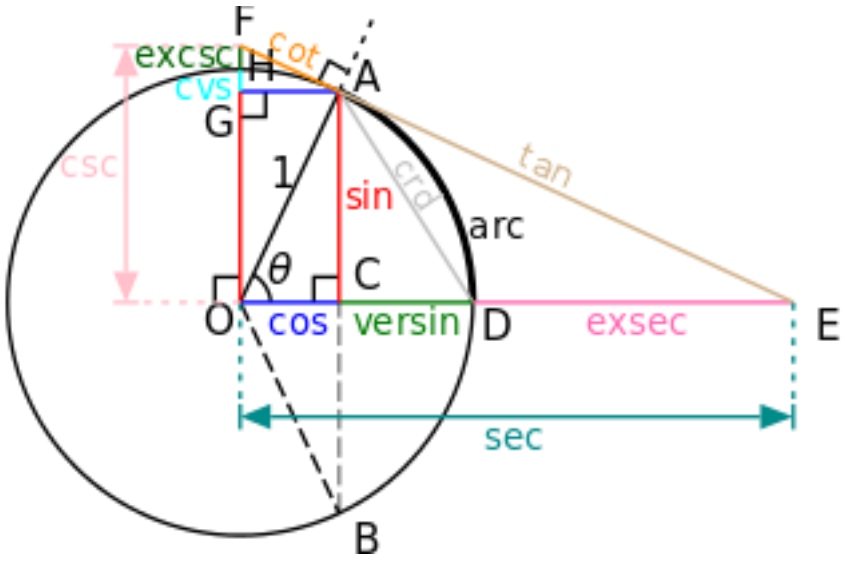

Рисунок 3.1. Одиничне коло з зображенням окремих тригонометричних співвідношень.

На рисунку 3.1 наведено одиничне коло із зображенням рідкісних тригонометричних відносин. Формула гаверсинуса є специфічним прикладом загального правила сферичної тригонометрії, а саме закону гаверсинусів, який стосується сторін та кутів у сферичних трикутниках [27]. Перший довідник гаверсинусів було представлено Джеймсом Ендрю у 1805 році.

$$a = \sin^2\left(\frac{|\phi_1 - \phi_2|}{2}\right) + \cos\phi_1 \cos\phi_2 \sin^2\left(\frac{|\lambda_1 - \lambda_2|}{2}\right),$$

$$c = 2\arctan\left(\sqrt{\frac{a}{1-a}}\right),$$

$$d = Rc$$

У цій формулі φ вказує на широту, λ – на довготу, R – на радіус Землі (у середньому це 6,371 км), а d – це відстань, яку ми шукаємо. При розрахунках відстані висота польоту не враховується, оскільки у розглянутих ситуаціях вона є незначною порівняно із загальною дистанцією перельоту.

3.2 Математична модель задачі планування місій гібридних комплексів

Загальна схема розв'язування задачі полягає у розбитті множини цілей і призначенні їх наявним базам з подальшим таким їх розбиттям на підмаршрути, які починаються і закінчуються на призначеній базі і при цьому не порушують обмеження на польотний ресурс дрона.

Нагадаємо, що розбиттям довільної множини називається сімейство непустих диз'юнктних її підмножин, об'єднання яких дає всю множину; такі підмножини називаються блоками розбиття (БР).

Розглядається планарна задача, тобто цілі і бази знаходяться на площині (двовимірне розташування на місцевості). Вхідні дані задачі подано у табл. 3.1.

Таблиця 3.1. Вхідні дані задачі

Параметр	Пояснення
B	множина заданих на маршруті баз транспортного засобу
b	кількість баз на маршруті
Z	множина цілей
n	кількість цілей
T	ресурс дрона у часовому вимірі (час у польоті без поповнення)
Θ	час відновлення ресурсу (обслуговування) дрона
v	швидкість дрона
w	швидкість транспорту
D	матриця відстаней між цілями, базами та між цілями і базами

Задача полягає у мінімізації часу виконання всієї місії шляхом оптимізації загального маршруту дрона. Цей маршрут складається з підмаршрутів, які пов'язані з вибором баз для виконання завдань та з урахуванням ресурсу дрона, а також із визначенням графіку руху транспорту та дрона, що враховує рух до баз на маршруті та польоти з них для відвідування цілей, а також процес поповнення ресурсу. Важливо зауважити, що час прибуття транспорту на

кожну базу не фіксований, його слід визначити під час планування місії. Іноді може бути доцільним, щоб не всі бази використовувалися, тобто дрон не буде злітати з деяких баз.

Пронумеруємо елементи множин *Z* та *B*.

Верхній рівень розбиття множини цілей, пов'язаних з базами, можна подати матрицею $X=(x_{ij})_{b\times n}$, у якій елементи $x_{ij}\in\{0,1\}$ визначаються так:

$$x_{ij}=\begin{cases}1, \text{ якщо ціль } j \text{ призначена на базу } i,\\ 0 \text{ в іншому разі,}\end{cases}$$

де $i=1,\ldots,b$, $j=1,\ldots,n$.

Обмеження на ресурс дрона може призвести до необхідності виконання декількох вильотів з певної бази (підмаршрутів) для обстеження цілей.

Щоб відобразити цю ситуацію, введемо тензор $U=(u_{lj}{}^{i})_{n\times n\times b}$, де

$$u_{lj}{}^{i}=\begin{cases}1, \text{ якщо на базі } i \text{ ціль } j \text{ включена в підмаршрут } l,\\ 0 - \text{ в іншому разі,}\end{cases}$$

де $i=1,\ldots,b$, $j=1,\ldots,n$, $l\in\{1,\ldots,n\}$.

Тоді обмеження задачі можна подати так:

$$\sum_{i=1}^{b} x_{ij}=1,\ j=1,\ldots,n, \tag{3.1}$$

$$\sum_{l=1}^{n} u_{lj}{}^{i}=x_{ij},\ j=1,\ldots,n,\ i=1,\ldots,b, \tag{3.2}$$

$$\frac{S_{il}(X,U)}{v}\le T,\ i=1,\ldots,b, l=1,\ldots,n. \tag{3.3}$$

Тут $S_{il}(X,U)$ – це довжина підмаршруту як результат розв'язування задачі комівояжера для підмаршруту *l* на базі *i*, якщо

$$\sum_{j=1}^{n} u_{lj}^{\ i} \neq 0,\ l=1,...,n,\ i=1,\ldots,b, \tag{3.4}$$

або ж $S_{il}(X,U)=0$ якщо база i не задіяна, тобто вильоти з неї не передбачаються планом, що формується.

Цільова функція задачі визначається часом виконання місії, який складається із часів задіяння транспорту та дрона.

Зробимо припущення, яке природним чином виконується на практиці (особливо, якщо поповнення ресурсу дрона – це заміна акумулятора): $\Theta < d(b_{i,i+1}) + d(b_{i+1,i+2}),\ i \in \{1,...,b\text{-}2\}$.

Наявність задіяної бази визначається нерівністю

$$\sum_{l=1}^{n}\sum_{j=1}^{n} u_{lj}^{\ i} \neq 0,\ \ i=1,\ldots,b. \tag{3.5}$$

Нехай I – максимальний номер задіяної бази при обраному розподілі цілей:

$$I = \max_{1\le i\le b}\{i : \text{виконується умова}\,(3.5)\}\ . \tag{3.6}$$

Тоді час руху транспорту t_T із врахуванням можливої наявності незадіяних баз визначається так:

$$t_T = \sum_{i=1}^{I-1} t_{i,i+1}, \tag{3.7}$$

де

$$t_{i,i+1} = \begin{cases} \max\{\frac{d(b_i, b_{i+1})}{w}, \Theta\}, \text{якщо для бази } i \text{ виконується нерівність (3.5)}, \\ \frac{d(b_i, b_{i+1})}{w} - \text{в іншому разі}. \end{cases} \quad (3.8)$$

Час $t_A^{\,i}$ використовування дрона на базі *i* обчислюватиметься тоді так:

$$t_A^{\,i} = (\sum_{l=1}^{n} \prod_{j=1}^{n} u_{lj}^{\,i} - 1)\Theta + \frac{S_{il}(X,U)}{v}. \quad (3.9)$$

Використовуючи зазначене, цільова функція задачі може бути подана так:

$$f(X,U) = \sum_{i=1}^{I_m}[t_{i,i+1} + (\sum_{l=1}^{n} \prod_{j=1}^{n} u_{lj}^{\,i} - 1)\Theta + \frac{S_{il}(X,U)}{v})]. \quad (3.10)$$

Рівняння (3.1) гарантує, що кожна ціль призначена на одну і тільки одну базу. Рівняння (3.2) визначає, що призначена на базу *i* ціль *j* включена в один із підмаршрутів *l* і не входить у підмаршрути на цій базі в іншому випадку. Нерівність (3.3) відображає обмеження на ресурс дрона. Рівняння (3.4) виділяє ті підмаршрути *l* на базі *i,* для яких розв'язується задача їх оптимізації. Умова, що база задіяна, тобто їй приписана хоча б одна ціль, подана нерівністю (3.5). Співвідношення (3.6)-(3.9) подають детально принципи обчислення компонент цільової функції задачі, яка визначає час виконання місії.

Задача планування місії гібридної системи "Дрон + ТЗ" полягає у пошуку мінімуму функції (3.10) з урахуванням обмежень та позначень (3.1)-(3.9) і належить до класу задач комбінаторної оптимізації.

Задача комбінаторної оптимізації, що виникає, може бути віднесена до задач оптимізації маршрутів дрона з багатьма візитами, як згадана в [52] задача MVDRP.

3.3 Математична модель задачі планування командних місій БПЛА

Введемо позначення:

$B=\{1,\ldots,b\}$ – множина пунктів (місць можливого базування), які потенційно можуть бути використаними в якості бази, b – кількість таких місць;

$N=\{1,\ldots,n\}$ – множина цілей, які слід відвідати, n – їх кількість;

$M=\{1,\ldots,m\}$ – множина наявних БПЛА, m – їх кількість;

d_{st} – відстань між цілями чи цілями і місцями можливого базування s,t, де $s,t \in B \cup N$;

c_i – вартість розміщення бази в пункті i, $i \in B$;

T_{ki} – оцінка часу обстеження k-м БПЛА цілі i, $i \in N$;

e_k – вартість затрат ресурсу k-го БПЛА на одиницю довжини шляху;

v_k – середня швидкість k-го БПЛА;

R_k – ресурс k-го БПЛА (вартість всього запасу палива чи заряду акумулятора).

Включення перельоту з точки (бази, цілі) i у точку (ціль, базу) j для БПЛА k задаватиметься змінними x_{ijk}:

$$x_{ijk} = \begin{cases} 0, \text{ переліт не виконується}, \\ 1, \text{ переліт виконується}, \end{cases}, \; i, j \in B \cup N, k \in M. \qquad (3.11)$$

Задача полягає у мінімізації загальної вартості місії і може бути поданою так:

знайти

$$\min \sum_{k \in M} \sum_{i \in B \cup N} \sum_{j \in B \cup N} e_k d_{ij} x_{ijk} + \sum_{k \in M} \sum_{i \in N} \sum_{j \in B \cup N} e_k v_k T_{kj} x_{ijk} \qquad (3.12)$$

за обмежень

$$\sum_{k \in M} \sum_{i \in B \cup N} x_{ijk} \le 1, \; j \in N; \qquad (3.13)$$

$$\sum_{k\in M}\sum_{j\in B\cup N} x_{ijk} \le 1,\ \ i \in N; \tag{3.14}$$

$$\sum_{i\in B}\sum_{j\in N} x_{ijk} \le 1,\ \ k \in M; \tag{3.15}$$

$$\sum_{i\in B}\sum_{j\in N} x_{jik} \le 1,\ \ k \in M; \tag{3.16}$$

$$u_i^k - u_j^k + x_{ijk}\sum_{s\in B\cup N}\sum_{t\in B\cup N} x_{stk} \le \sum_{s\in B\cup N}\sum_{t\in B\cup N} x_{stk} - 1,\ \ u_l^k \in \mathbf{R},\ i, j \in B\cup N, k \in M; \tag{3.17}$$

$$\sum_{i\in B}\sum_{j\in B} x_{jik} = 0,\ \ k \in M; \tag{3.18}$$

$$d_{ij} = \infty,\ \ i, j \in B; \tag{3.19}$$

$$\sum_{i\in B\cup N}\sum_{j\in B\cup N} e_k d_{ij} x_{ijk} + \sum_{i\in N}\sum_{j\in B\cup N} e_k v_k T_{kj} x_{ijk} \le R_k,\ \ k \in M; \tag{3.20}$$

$$x_{ijk} \in \{0,1\}, i, j \in B\cup N,\ \ k \in M; \tag{3.21}$$

$\mathbf{R}$ – множина дійсних чисел. (3.21)

Цільова функція (3.12) визначає сумарні затрати від планування вибору баз для БПЛА разом із побудовою маршрутів для обльоту цілей та затраченого часу на їх обслуговування.

Кожен БПЛА відвідує певну кількість цілей, але прибуття та відправлення від кожної цілі має здійснитися лише один раз одним БПЛА. Одиничне значення у формулі (3.13) означає, що лише один БПЛА залишає ціль *j*, а (3.14) – що лише один БПЛА прибуває до цілі – або ж нулю, якщо БПЛА *k* не задіяний. Формула (3.15) задає умову вильоту БПЛА з однієї із можливих баз, а формула (3.16) – умову повернення на одну із таких баз; знову ж, якщо БПЛА задіяний у варіанті розв'язку, то маємо рівність. Формула (3.17) задає умову уникнення підциклів у маршруті кожного БПЛА, що робить матрицю розв'язків несиметричною. Вимога не перелітати від однієї можливої бази безпосередньо до іншої відображена в формулах (3.18)–(3.19). У формулі (3.20) подано

урахування обмежень на польотні ресурси БПЛА. Нарешті, формули (3.21)–(3.22) задають області визначення змінних задачі.

3.4 Висновки до розділу

У цьому розділі наведено детальні математичні моделі двох ключових задач дослідження – планування місії гібридних систем та планування місій команд БПЛА. Для задачі планування місії гібридної системи побудовано математичну модель з урахуванням усіх основних компонентів: дрон, транспортний засіб, цілі, бази, обмеження ресурсу дрона. Формалізовано процес розбиття множини цілей по базах і підмаршрутам з обмеженнями.

Для задачі планування місії групи БПЛА побудовано модель, що враховує агентів та їх характеристики, цілі, бази, обмеження. Формалізовано цільову функцію та обмеження, визначено змінні задачі.

Розроблені математичні моделі дозволили формалізувати та уточнити постановки задач планування місій гібридних систем та команд БПЛА. Ці моделі є важливою базою для подальшої розробки ефективних обчислювальних методів розв’язання даних оптимізаційних задач. Запропоновані підходи до формалізації можуть бути використані і для інших споріднених задач планування місій із групами автономних агентів.

РОЗДІЛ 4. РОЗРОБКА АЛГОРИТМІВ ПЛАНУВАННЯ БЕЗПІЛОТНИХ МІСІЙ

4.1 Вступні положення

У цьому розділі подано розроблені алгоритми для розв'язування задач планування місій гібридної системи та команди БПЛА. Для задачі гібридної системи запропоновано жадібний алгоритм, в якому дрон розглядається як агент з власними характеристиками. Описано особливості реалізації переходів між вершинами графа для цього алгоритму. Також для задачі планування місії гібридної системи розроблено алгоритм детермінованого локального пошуку на основі процедур 2-замін та представлені ключові аспекти реалізації цього алгоритму. З використанням цих двох алгоритмів у якості метаевристик розроблено спеціалізований алгоритм оптимізації мурашиними колоніями, що враховує особливості постановки задачі планування місій систем «Дрон + ТЗ».

Для задачі планування місій команди БПЛА запропоновано жадібний алгоритм, в якому кожен БПЛА є агентом з власними характеристиками. Описано особливості реалізації переходів між вершинами графа для кожного БПЛА з урахуванням маршрутів інших учасників команди. Також для задачі планування місії команди БПЛА розроблено алгоритм детермінованого локального пошуку на основі процедур 2-замін та представлені ключові аспекти реалізації цього алгоритму з урахуванням підмаршрутів кожного агента. З використанням цих двох алгоритмів у якості метаевристик розроблено спеціалізований алгоритм оптимізації мурашиними колоніями, що враховує особливості постановки задачі планування місії команди БПЛА.

Запропоновані алгоритми дозволять ефективно розв'язувати складні задачі планування безпілотних місій різних типів. Дослідження отриманих результатів та аналіз ефективності розроблених алгоритмів подані у наступних розділах.

4.2 Основні поняття та позначення

Для опису комбінаторної постановки будуть використані такі позначення:

$||X||$ – потужність довільної множини X;

$O(x)$ –довільний окіл точки $x \in X$.

Означимо досліджуваний простір як вибрану множину (назвемо її твірною), між елементами якої існують певні співвідношення. Булеаном 2^X довільної множини X називається множина всіх її підмножин [19].

Нехай на множині X тим чи іншим чином описана система околів O, тобто для довільного $x \in X$ визначено набір множин:

$$o^{\sigma}(x) \subseteq 2^X, \sigma \in I, \tag{4.1}$$

де I - множина індексів околів.

Дискретний простір – це математичний простір, елементи якого можна відділити один від одного. В такому просторі елементи не мають безперервних переходів між собою, на відміну від елементів неперервного простору. Найбільш поширеним прикладом дискретного простору є множина цілих чисел, де між будь-якими двома послідовними числами немає інших чисел. Іншими словами, це такий простір, у якому твірна множина X є скінченною або нескінченною без граничних точок, які також називають точками конденсації, дотику або накопичення.

Зокрема, застосовуються наступні види околів:

- метричні;
- топологічні;
- алгоритмічні або дескриптивні.

Метричні або топологічні добре відомі в математиці. Процедура утворення алгоритмічних околів має бути певним чином описана, наприклад 2-заміни в задачі комівояжера [33].

Дескриптивні визначаються задаються в описовій чи аналітичній формі (наприклад шляхом явного переліку для кожного $x \in X$ елементів, які належать його околам).

Розглянемо метричний простір *X* – тобто такий, на якому введено метрику *d(x, y)*.

Метричним околом із радіусом $\rho>0$ називають множину:

$$L_\rho(x) = \{y \in X : d(x,y) \le \rho\}. \tag{4.2}$$

Простір *(X, O)* називається дискретним, якщо

$$\forall x \in X \exists o^\sigma(x) \in O : o^\sigma(x) = \{x\}. \tag{4.3}$$

Для кожної точки дискретного простору існує окіл, що складається виключно із цієї точки – таку точку називають ізольованою.

Простір *X*, якщо він складається лише із ізольованих точок називається дискретним.

Варіант розв'язку $x_* \in S$ називається субоптимальним розв’язком задачі, якщо:

$$f(x_*) \le f(y), \forall y \in S. \tag{4.4}$$

Якщо *S* збігається з околом точки x, тобто $S = o^\sigma(x)$, де $o^\sigma(x)$ – деякий окіл точки *x*, то такий субоптимальний розв'язок називається локально оптимальним, або локальним.

Уведені означення дозволяють формально визначити поняття глобального екстремуму.

Якщо умова (4.4) виконується для $S = X$, то точка x_* називається глобальним, або точним, розв’язком задачі. Якщо система околів така, що будь-який локально оптимальний розв'язок задачі глобальний, то кажуть, що така система околів є точною.

Часто під ЗКО розуміють проблему пошуку екстремумів за даної цільової функції, коли X – комбінаторний простір.

Комбінаторним простором вважають сукупність комбінаторних об'єктів певного типу, утворених із елементів заданої скінченної множини (твірна множина), хоча при цьому формального означення не наводять [33].

Базисними околами довільної точки $x \in X$ будемо називати наступну множину:

$$B_x = \{o^\tau(x) \in O : \left\|o^\tau(x)\right\| > 1 \,\&\, \nexists \gamma : 1 < \left\|o^\gamma(x)\right\| < \left\|o^\tau(x)\right\|\} \qquad (4.5)$$

Комбінаторним простором назвемо дискретний локально скінченний у комбінаторному розумінні простір, який має не більш ніж зліченну кількість елементів.

Задача називається задачею комбінаторної оптимізації (ЗКО), якщо простір її розв’язків X є комбінаторним [33].

4.3 Визначення початкового розміщення БПЛА

Початкове розміщення БПЛА базується на введенні поняття нульової бази – віртуальної точки, у якій розміщуються усі наявні БПЛА до початку роботи алгоритму. Необхідна лише одна така база, означимо множину D, що складається виключно з неї. Ця база є частиною графу задачі, що складається з наступних компонент:

- початкових та нульової баз;
- цілей;

- ребер, що попарно з'єднують усі цілі та початкові бази між собою;
- ребер, що з'єднують нульову базу з початковими.

Повернення до нульової бази та переміщення з нульової бази безпосередньо до цілей заборонені. Усі ребра, що виходять з нульової бази мають нульову довжину.

$$d_{st} = 0, \text{ де } s \in D, t \in B \tag{4.8}$$

Переміщенням з нульової бази до цілі вважається послідовне переміщення до найближчої до цілі початкової бази (довжина такого переміщення нульова) та переміщення з цієї бази до цілі. Таким чином, довжина переміщення визначається за формулою:

$$d_{st} = min\{d_{kt}\}, \text{ де } s \in D, k \in B, t \in N \quad . \tag{4.9}$$

Початкове розміщення БПЛА є частиною отриманого розв'язку, оскільки перельоти між нульовою базою та ціллю (відповідно, початкове розміщення кожного БПЛА) є предметом оптимізації для загального алгоритму на поданому графі задачі з урахуванням (4.8)-(4.9).

4.4 Ключові аспекти реалізованого жадібного алгоритму

У рамках розробленого жадібного алгоритму кожен БПЛА розглядається як самостійний агент, що характеризується:

- початковим запасом ходу;
- залишковим запасом ходу (відстань, яку БПЛА ще може пролетіти, починаючи з поточного стану, рівний початковому при вильоті з бази);
- пройденим шляхом;
- поточною позицією (одна з вершин).

На початку роботи алгоритму всі БПЛА розміщуються у «нульовій» точці, описаної у розділі 4.3 – умовна точка, однокрокові перельоти з якої

безпосередньо на базу заборонені, а відстань до цілі визначається як мінімальна відстань від даної цілі до бази.

Такий підхід дозволяє зробити задачу початкового розміщення БПЛА частиною задачі їх маршрутизації. Отже, перехід БПЛА *k* в один крок з «нульової» точки *o* до цілі *i* буде здійснюватися наступним чином:

- у якості позиції БПЛА встановлюється ціль *i*;
- у шлях БПЛА додається найближча до цілі база та сама ціль;
- від залишкового запасу ходу віднімається найменша з відстаней від цілі до бази;
- ціль помічається відвіданою.

Переходи між цілями та базами (*i* – початкова точка, *j* – кінцева) здійснюються за аналогічною схемою, проте без проміжних точок:

- у якості позиції БПЛА встановлюється точка *j*;
- у шлях БПЛА додається точка *j*;
- від залишкового запасу ходу віднімається відстань між точками *i* та *j*;
- якщо *j* – ціль, позначається відвіданою.

У разі переходу на базу здійснюється заміна блоків живлення, залишковий запас ходу встановлюється на рівні початкового. Слід зазначити також, що при здійсненні переходів слід перевіряти досяжність інших вершин після його здійснення, інакше можлива ситуація, за якої допустимий розв’язок не буде знайдено.

4.5 Загальна схема алгоритму детермінованого локального пошуку

Алгоритми детермінованого локального пошуку - це сім'я ітераційних методів, заснована на частковому перебиранні варіантів на кожній ітерації серед точок околу поточної точки, тобто серед сусідніх до неї. В алгоритмах цього типу замість повного перебору застосовується спрямований локальний перебір у підмножинах варіантів, які називаються околами. Цим пояснюється їх назва – алгоритми локального пошуку (Local Search). У сфері КО алгоритми ЛП

мають давню історію через свою наочність і високу ефективність. Наприклад, перший алгоритм локального пошуку для задачі комівояжера був запропонований ще в 1956 р., а локальний пошук для задачі розміщення обладнання був розроблений у 1962 р.

Загальна схема ЛП у задачах мінімізації виглядає наступним чином. Починаючи з деякого припустимого розв'язку задачі, новий розв'язок із кращим значенням цільової функції шукають у його околі. Якщо такий розв'язок знайдено, то він приймається і пошук поліпшення розв'язку далі здійснюється вже в його околі і т. д. Алгоритм завершується, коли досягнуто локального оптимуму, тобто коли в околі поточного розв'язку немає ніякого іншого варіанта з меншим значенням цільової функції.

Загальна схема алгоритмів детермінованого локального пошуку може бути подана таким чином:

- генерація початкового припустимого розв'язку x, який обираємо як поточний варіант;
- чергова ітерація – формуємо окіл $O(x)$ поточного варіанта й точно чи наближено знаходимо елемент $y \in O(x)$, який є субоптимальним розв'язком у цьому околі. Якщо $y \neq x$, то знайдений елемент оголошується черговим поточним варіантом і починається чергова ітерація, інакше – наступний пункт;
- завершення роботи алгоритму: x – локальний розв'язок, якщо на останній ітерації здійснюється вичерпний пошук в околі.

4.6 Ключові аспекти реалізованого алгоритму ДЛП

Загалом, принциповими моментами реалізації конкретних алгоритмів локального пошуку є:

- визначення околів $O(x)$;
- генерація чергової точки $y \in O(x)$;
- критерій завершення перегляду точок у поточному околі та переходу до наступного;

- спосіб обчислення величини зміни цільової функції при переході до нового поточного варіанта;
- критерій завершення;
- формування початкового наближення.

Використано відомий алгоритм 2-замін Ліна – один з сімейства алгоритмів k-замін (k-exchange), що базуються на процедурі вилучення *k* ребер і додавання *k* нових. Для побудови сусідніх варіантів (околу) до поточного маршруту було запропоновано використовувати такі процедури:

- транспозиції міст;
- вставка міста;
- переміщення фрагмента шляху;
- заміна k ребер.

Схема алгоритму 2-opt Ліна, що базується на процедурі 2-заміни, поданій на рисунку 4.2. Неважко показати, що обчислювальна складність алгоритму – $O(n^2)$.

```
procedure Execute_LS (x)
foreach підмаршруту (між вильотом та прильотом з/на базу) as route do
    :start_again:
    foreach ціль у підмаршруті as i do
        foreach ціль у підмаршруті після i as j do
            new_route += маршрут_між(route.start, i);
            new_route += обернений_маршрут(маршрут_між(i+1, j));
            new_route += маршрут_між (j+1:route.end);
            if маршрут_між (new_route) < довжина_маршруту(route) then
                route = new_route;
                goto start_again;
            endif
        endforeach
    endforeach
endforeach
end
```

Рисунок 4.2 – псевдокод алгоритму 2-opt

4.7 Загальна схема алгоритму ОМК для задачі планування командних місій.

Розроблено два спеціалізовані алгоритми, що базуються на методі оптимізації мурашиними колоніями (ОМК) та методі детермінованого локального пошуку (ДЛП) [2]. В алгоритмах ОМК формується спеціальна модель задачі, що розв'язується, тому вони належать до класу моделе-орієнтованих методів. Модель задачі подається у вигляді зваженого графа *G(V,E)*, де $v_i \in V, i = 1, \ldots, n + k$ – вершини, що відповідають компонентам розв'язку, а $e_{ij} \in E, e_{ij} = (v_i, v_j), v_i, v_j \in V$ – ребра, які відповідають можливим з'єднанням (переходам) між відповідними вершинами (містами). Для кожного ребра визначена функція вартості з'єднання, що відповідає відстані по поверхні між вершинами, з’єднаними даним ребром.

На кожному кроці алгоритму для будь-якої вершини $i \in V$ може бути побудована множина сусідніх вершин N_j.

Евристична інформація η_{ij} – це числове значення, що не залежить від знайдених на попередніх кроках розв'язків і відображає ступінь бажаності включення в побудований фрагмент розв'язку того чи іншого нового ребра графа моделі $e_{ij} \in E$. Евристичні значення базуються на апріорній інформації, що відображає умови конкретної задачі та надається джерелом, відмінним від мурах.

Рівень феромону (феромонний слід) – τ_{ij}, що відповідає ребру $e_{ij} \in E$, – це додатне число, яке показує, наскільки часто мурахами використовувалося це ребро на попередніх кроках чи при формуванні повного розв'язку. Феромонні сліди виконують роль довготривалої пам'яті для мурах щодо всього процесу пошуку.

Отже, основні компоненти обчислювальної схеми мурашиних алгоритмів такі:

- модель задачі, що подається спеціальним графом;
- феромонні значення;
- евристична інформація;
- пам'ять (локальна та глобальна).

Загальну обчислювальну схему алгоритмів ОМК можна подати так, як на рисунку 4.3.

```
procedure ACO (x)
    ініціалізація_алоритму;
    while критерій_завершення_не_задоволений do
        формування_популяції_мурах;      {поточне покоління}
        foreach мураха_з_популяції do      {життєвий цикл мурахи}
            ініціалізація_мурахи;
            M = оновлення_пам'яті_мурахи;
            while поточний_стан ≠ повний_розв'язок do
                A = локальна_ матриця_мурашиних_маршрутів;
                сформувати_множину припустимих_вершин;
                p  = обчислити_ймовірність_переходів(A, M, П);
                наступний_стан = правило_прийняття_рішення(p, П);
                перейти_в_наступний_стан(наступний_стан);
                if онлайнове_покрокове_оновлення_феромону then
                    відкласти_феромон_на_відвіданій_дузі;
                    поновити_ матрицю_мурашиних_маршрутів_A;
                endif
                M = оновити_внутрішній_стан;
            endwhile
            if онлайнове_відстрочене_оновлення_феромону then
                foreach відвіданої_дуги_побудованого_розв'язку do
                    відкласти_феромон_на_відвіданій_дузі;
                    оновити_матрицю_мурашиних_маршрутів_A;
                endforeach
            endif
            завершити діяльність;
        endforeach
        випаровування_феромону;
        оновлення_рекорду (x);
        дії_Демона;              {необов'язково}
    endwhile
end
```

Рисунок 4.3. Загальна обчислювальна схема алгоритму ОМК

Детальніше розглянемо правила переходу до наступної вершини та процес обчислення ймовірностей переходів. Стани задачі визначаються в термінах скінченних послідовностей $y = (v_{s_1}, v_{s_2}, \dots), v_{s_r} \in V$ елементів V (або, що рівносильно, E), які на всіх проміжних кроках мурахи є фрагментами розв'язку задачі оптимізації. Якщо Y – множина всіх можливих послідовностей, то множина Y^{∞} усіх (під)послідовностей, які задовольняють обмеження $\Pi = \Pi(V, E, t)$, є підмножиною Y: $Y^{\infty} \subseteq Y$, а її елементи визначають припустимі стани задачі. Нехай на певному кроці мураха k побудувала фрагмент розв'язку y, останньою компонентою якого є вершина $i \in V$, тобто вона перебуває в цій вершині: $y = (\dots, i)$. Тоді мураха може переміститися до будь-якої вершини j із множини припустимих сусідніх вершин N_i^k, що визначаються як $N_i^k = \{j: j \in N_i \wedge (y, i) \in Y^{\infty}\}$, де N_i – множина всіх сусідніх до i вершин графа задачі [12], [17]. Вибір наступної вершини відбувається за псевдовипадковим пропорційним правилом. Введемо новий параметр $p_0 \in [0,1]$, кожна мураха переходить з вершини $i \in V$ до вершини $j \in N_i^k$ з імовірністю p_0, j визначається наступним чином: $j = arg\ max\left\{\tau_{ir}^{\alpha}(t)\eta_{ir}^{\beta}, r \in N_i^k\right\}$, а з імовірністю $1 - p_0$ обирається вершина за правилом колеса рулетки з використанням імовірності p_{ij}^k:

$$p_{ij}^k = \frac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\tau_{ir}^{\alpha}(t)\eta_{ir}^{\beta}(t)}\ p_{ij}^k = \frac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{r \in N_i^k}\tau_{ir}^{\alpha}(t)\eta_{ir}^{\beta}(t)}. \quad (4.10)$$

Відкладання та випаровування феромонів відбувається за такою формулою:

$$\tau_{ij}(t+1) = \rho\tau_{ir}(t) + \frac{(1-\rho)}{f_{min}^0},$$

де ρ – коефіцієнт випаровування, що лежить в межах від 0 до 1, а f_{min}^{0} – найкраще значення цільової функції на початковій популяції мурах.

Нижня та верхня межа феромонів визначаються за такими формулами [10]:

$$\tau_{max} = \frac{1}{\rho f_{min}^{0}},$$

$$\tau_{min} = \left[\frac{\tau_{max}\left(1-\sqrt[n]{0.05}\right)}{\left(\left(\frac{n}{2}-1\right)\sqrt[n]{0.05}\right)}\right],$$

де $n = |V|$.

Коригування матриці феромонів відбувається наступним чином:

$$a' = min\{a, \tau_{max}\}, a' = max\{a', \tau_{min}\},$$

де a – елемент матриці феромонів.

Для уникнення стагнації здійснюється скидання значень матриці феромонів у початковий стан, якщо за визначену кількість ітерацій не відбулося покращення найкращого розв’язку. Для розв’язання наведеної задачі розроблено алгоритм, що враховує такі аспекти:

- вибір підмножини БПЛА, що мають бути задіяними;
- вибір початкової та кінцевої баз для кожного задіяного БПЛА;
- формування плану місії, що мінімізує загальні затрати, необхідні для обстеження усіх наявних цілей.

Початкове розміщення БПЛА відбувається завдяки введенню поняття нульової бази – точки, у якій розміщуються усі наявні БПЛА до початку роботи алгоритму. Необхідна лише одна така база, означимо множину *D*, що складається виключно з неї. Ця база є частиною графу задачі, що складається з наступних компонент:

- початкових та нульової баз;
- цілей;
- ребер, що попарно з’єднують усі цілі та початкові бази між собою;
- ребер, що з’єднують нульову базу з початковими.

Повернення до нульової бази та переміщення з нульової бази безпосередньо до цілей заборонені. Усі ребра, що виходять з нульової бази мають нульову довжину.

$$d_{st} = 0, \text{ де } s \in D, t \in B \tag{4.11}$$

Переміщенням з нульової бази до цілі вважається послідовне переміщення до найближчої до цілі початкової бази (довжина такого переміщення нульова) та переміщення з цієї бази до цілі. Таким чином, довжина переміщення визначається за формулою:

$$d_{st} = min\{d_{kt}\}, \text{ де } s \in D, k \in B, t \in N \quad . \tag{4.12}$$

Початкове розміщення БПЛА є частиною отриманого розв’язку, оскільки перельоти між нульовою базою та ціллю (відповідно, початкове розміщення кожного БПЛА) є предметом оптимізації для загального алгоритму на поданому графі задачі.

Для розв’язування сформульованої задачі використано модифікований макс-мін алгоритм мурашиних систем з покроковою побудовою розв’язку. Кожна мураха, що вибирає наступну вершину графа задачі, робить її недоступною для подальшого відвідування після переходу. Для всіх мурах на кожному кроці відбувається наступна послідовність дій:

- формування підмножини припустимих вершин, які може відвідати мураха;
- обчислення ймовірності переходів від поточної вершини і до всіх допустимих вершин;

- вибір допустимої вершини і перехід до неї.

Кожна мураха формує лише частковий розв’язок задачі, а їх сукупність дає повний розв’язок. Через це на кожній ітерації формується набір часткових розв’язків, з яких обирається найкращий, на ньому відкладається феромон. На наступних етапах відбуються такі дії:

- випаровування феромонів;
- оновлення допустимих нижньої та верхньої межі феромону;
- оновлення матриці феромонів відповідно нижньої та верхньої межі.

На рис. 4.4 наведено обчислювальну схему розробленого алгоритму для розв'язування поставленої задачі. Загальна процедура, що керує процесом розв'язування складається з:

- запуску жадібного алгоритму для визначення Q – початкового наближення розв'язку для застосування у формулі обчислення феромонного сліду;
- початкового розміщення агентів;
- підбору параметрів;
- запуску з обраними параметрами.

```
procedure UAVRP (x)
  запуск жадібного алгоритму;
  початкове розміщення агентів у нульовій базі;
  while не відвідано всі цілі do
      сформувати набір допустимих цілей для усіх агентів з урахуванням
запасу ходу;
      if нема допустимих then
          сформувати набір допустимих цілей з відвідуванням бази;
      endif
      перемістити агента, що відповідає найкоротшому переміщенню до
обраної цілі;

      foreach кожного набору параметрів do
          foreach кожного джерела рандомізації do
              запуск мурашиного алгоритму зі строгим часовим
обмеженням;
          endforeach
      endforeach

      вибір параметрів, що показали найменше значення цільової функції;
      повний запуск з обраним набором параметрів;
  endwhile
end
```

Рисунок 4.4. Обчислювальна схема алгоритму для розв'язування задачі

Основна частина мурашиного алгоритму подана на рис. 4.5.

```
procedure UAVRP_ACO (x)
  while не досягнуто ліміту часу чи ітерацій do
      ініціалізувати нову ітерацію;
      початкове розміщення агентів у нульовій базі;

      while не відвідано всі цілі do
          знайти допустимі перельоти до цілей;
          foreach невідвіданої цілі do
              перевірити, чи достатньо ресурсу для відвідування;
              перевірити, чи буде достатньо ресурсу для повернення на
базу;
          endforeach
          if є допустимі опції then
                  обрахувати ймовірність здійснення кожного з цих
перельотів;
              виконати переліт згідно розподілу ймовірностей;
          else
                  перемістити усіх агентів на відповідні найближчі бази;
          endif
          оновити запас ходу агента;

      if з фіксованою ймовірністю then
              запустити локальний пошук (дії Демона);
      endif

      if виникла помилка then
          ініціалізувати нову ітерацію;
              продовжити з початку циклу;
      endif
```

```
    if найкращий результат then
                записати результат (далі - score) та маршрути;
        score = результат;
                оновити Q - чисельник для обчислення феромону;
    endif
    здійснити випаровування з урахуванням обмежень зверху та знизу;
    додати Q / score до кожного перельоту з результату;
  endwhile
end
```

Рисунок 4.5. Обчислювальна схема розробленого алгоритму ОМК

Кожен агент (мураха, що відповідає БПЛА) має ймовірність здійснення переміщення під час кожної ітерації алгоритму, тож безпосередня взаємодія агентів відбувається протягом усієї роботи алгоритму. Подана процедура викликається як у рамках модифікованого ОМК, так і у модифікованому ДЛП у якості основної частини алгоритму.

4.8 Загальна схема алгоритму ОМК для задачі планування місій гібридних систем.

Для розв'язування поданої задачі було розроблено три алгоритми: жадібний, на базі методу оптимізації мурашиними колоніями (ОМК) та методу детермінованого локального пошуку (ДЛП). Зокрема, розв'язки, отримані за допомогою жадібного алгоритму використані як початкові наближення для методу ДЛП, а також для приблизної оцінки оптимальної вартості місії для корекції випаровування феромону у рамках спеціалізованого алгоритму на базі ОМК. Також, алгоритм, що базується на ДЛП використовується для проміжного покращення результатів, що отримуються у ході роботи спеціалізованого алгоритму ОМК.

Модель задачі подається у вигляді зваженого графа $G(V,E)$, де $v_i \in V$, $i=1,\ldots,n+k$ – вершини, що відповідають компонентам розв'язку; $e_{ij} \in E$, $e_{ij}=(v_i, v_j)$

$\in V$ – ребра, які відповідають допустимим з'єднанням між відповідними вершинами (цілями).

Для кожного ребра визначена функція вартості з'єднання, що відповідає геодезичній відстані між вершинами, з’єднаними даним ребром.

На кожному кроці поданих алгоритмів для будь-якої вершини $i \in V$ будується множина сусідніх вершин N_j.

Евристична інформація, позначена як η_{ij}, є числовим значенням, яке не залежить від результатів, отриманих на попередніх етапах розв'язку, і відображає ступінь бажаності включення ребра графа моделі, позначеного як $e_{ij} \in E$, до побудованого рішення. Ці евристичні значення ґрунтуються на апріорній інформації, що відображає умови конкретної задачі та надається джерелом, відмінним від мурах.

Рівень феромону, позначений як τ_{ij}, асоціюється з кожним ребром $e_{ij} \in E$. Це додатне число, яке вказує на частоту використання цього ребра мурахами на попередніх кроках або під час формування повного розв'язку. Феромонні сліди виконують функцію довготривалої пам'яті для мурах стосовно всього процесу пошуку.

Таким чином, ключовими елементами обчислювальної схеми, що використовується у мурашиних алгоритмах, є такі компоненти:

- модель поставленої задачі, яка подається у вигляді спеціального графа;
- значення феромонів, які відіграють важливу роль у визначенні шляхів та рішень;
- евристична інформація, яка впливає на процес прийняття рішень мурахами;
- пам'ять, яка поділяється на локальну і глобальну складові, і забезпечує збереження та обмін інформацією між мурахами.

Розроблений жадібний алгоритм базується на схожій моделі, але не враховує евристичну інформацію та пам'ять, послуговуючись виключно інформацією про розташування вершин графу, а відтак відстанню між цілями

та базами. Оскільки алгоритм є детермінованим, використання багатьох агентів не має сенсу, тож задіяно лише одного агента.

procedure *Greedy*

розмістити агента у початковій точці;

поки не відвідано всі цілі:

обчислити довжини допустимих шляхів до невідвіданих цілей;

вибрати найкоротший шлях;

якщо ресурс агента недостатній для відвідування цілі і повернення на базу:

перемістити агента на попередю базу;

перемістити агента у початкову точку;

інакше:

перемістити агента найкоротшим шляхом, позначити ціль як відвідану;

end

Рисунок 4.6. Загальна обчислювальна схема жадібного алгоритму

Загальну обчислювальну схему алгоритмів ОМК можна подати таким чином:

procedure *ACO*
ініціалізувати популяцію мурах у початковій точці;
отримати наближення розв'язку за допомогою жадібного алгоритму;
встановити параметри алгоритму (кількість мурах, швидкість випаровування феромонів і інші);
для кожної ітерації алгоритму:
    поки не відвідано всі цілі:
    для кожної мурахи:
        якщо відсутні шляхи, після яких у мурахи вистачить ресурсу на повернення:
            перемістити мураху на попередню базу;
            перемістити мураху у початкову точку;
        інакше:
            обчислити шляхи на основі феромонів та евристичної інформації;
            вибрати шлях на підставі імовірності, вага залежить від феромону та
            евристичної інформації;
                перемістити мураху вибраним шляхом, оновити феромони на ньому;
    з фіксованою фмовірністю:
        запустити ДЛП для можливого покращення результату:
    оновити феромони на всіх шляхах відповідно до правила випаровування;
    знайти та записати найкращий шлях;
end

Рисунок 4.7. Загальна обчислювальна схема алгоритму ОМК

Розроблений спеціалізований ДЛП базується на алгоритмі метода вектору спаду з використанням оператора 2-opt замін та здійснюється для кожного отриманого фрагменту маршруту БПЛА від вильоту з бази до наступного повернення до неї. Тобто, оптимізація відбувається тільки для фрагментів шляху БПЛА, що починаються з бази, закінчуються тою самою базою, а між ними містять виключно цілі.

На початку роботи алгоритму усі наявні БПЛА розміщуються в точці, яку можна назвати "нульовою базою". Повернення до вихідної точки і безпосереднє пересування від неї до цілей заборонено. Усі ребра, які виходять з вихідної точки, мають нульову довжину: $d_{st}=0$, $s \in D$, $t \in B$.

Переміщення з нульової бази до цілі складається з двох кроків: спершу переміщення до найближчої початкової бази (яке має нульову довжину), а потім переміщення з цієї бази до цілі. Таким чином, довжина переміщення обчислюється згідно формулі: $d_{st}=\min\{d_{kt}\}$, $s \in D$, $t \in N$, $k \in B$.

Початкове розміщення БПЛА входить до складу отриманого розв'язку, оскільки оптимізація перельотів між нульовою базою та цільовою точкою, включаючи вибір початкового розміщення кожного БПЛА, є частиною загального алгоритму на поданому графі задачі.

У рамках жадібного алгоритму та алгоритму ОМК кожна мураха, що вибирає наступну вершину графа задачі, робить її недоступною для подальшого відвідування після переходу. Розроблений алгоритм ОМК є макс-мін алгоритмом з покроковою побудовою розв’язку.

4.9 Висновки до розділу

У розділі представлено детальний опис розроблених алгоритмів для розв'язування задач планування місій гібридних систем та команд безпілотних літальних апаратів.

Для задачі планування місій гібридної системи запропоновано жадібний алгоритм, в якому дрон розглядається як агент з власними характеристиками. Описано реалізацію переходів між вершинами графа з урахуванням обмежень

та вимог задачі. Також розроблено алгоритм детермінованого локального пошуку на основі процедур 2-замін та подано ключові аспекти його реалізації. На базі цих двох алгоритмів створено спеціалізований алгоритм оптимізації мурашиною колонією, адаптований до особливостей задачі планування місій гібридних систем.

Для задачі планування місій команди БПЛА також розроблено жадібний алгоритм, де кожен БПЛА є агентом з власними характеристиками. Описано реалізацію переходів між вершинами графа з урахуванням маршрутів інших БПЛА. Створено алгоритм детермінованого локального пошуку на основі 2-замін, адаптований до особливостей підмаршрутів кожного БПЛА. На основі цих двох алгоритмів розроблено спеціалізований алгоритм оптимізації мурашиною колонією для даної задачі.

Розроблені алгоритми дозволяють ефективно розв'язувати складні задачі планування місій різних типів безпілотників. Подальші дослідження можуть бути спрямовані на вдосконалення розроблених алгоритмів, а також на розширення їх можливостей для застосування до інших споріднених задач планування та оптимізації маршрутів.

РОЗДІЛ 5. ПИТАННЯ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

5.1 Вступні положення

У цьому розділі представлено обчислювальний блок, розроблений для реалізації алгоритмів планування місій гібридних систем та команд БПЛА. Описані використані технології та архітектурні рішення при програмній реалізації цього обчислювального блоку. Окрема увага приділятиметься протоколу обміну даними між його компонентами та зовнішніми системами, що дозволяє об’єднувати їх у ефективні програмні комплекси.

Також описано інтеграцію розробленого програмного забезпечення з популярними системами керування БПЛА, такими як UgCS, Mission Planner, APM Planner.

Для програмної реалізації обчислювального блоку використано мови програмування Python та C++, для розробки модуля візуалізації використано веб-представлення (HTML, CSS, JavaScript). Інтерфейс системи дозволяє задавати цілі та бази на мапі, коригувати та експортувати їх список, задавати набір доступних БПЛА та їх характеристики, задавати маршрут транспорту, вказувати параметри алгоритму. Після побудови плану місії його візуалізація доступна на мапі, де відображено усі заплановані маршрути БПЛА, цілі, бази, шлях транспорту. Окрема вкладка «Аналіз» дозволяє візуалізувати процес роботи алгоритму для спрощення подальших досліджень.

JSON (JavaScript Object Notation) був обраний як формат обміну даними між компонентами системи через його інтуїтивну зрозумілість та невелику надлишковість для даних, що описують цілі, бази, БПЛА та їх характеристики.

JSON є текстовим форматом для обміну даними між комп'ютерами, заснованим на читабельному для людини тексті. Він дозволяє описувати об'єкти та інші структури даних і використовується в основному для передачі структурованої інформації через мережу (серіалізації). JSON базується на двох основних структурах:

- набір пар "ключ-значення", реалізований в різних мовах як об'єкт, запис, словник тощо;
- впорядкований список значень, реалізований як масив, вектор, список тощо.

Це універсальні структури даних, підтримувані більшістю сучасних мов програмування, тому доцільно використовувати їх в JSON для обміну даними між різними середовищами.

Об'єкт – послідовність пар "ключ-значення", оточена фігурними дужками {}. Кожна пара відділена комою, а ключ від значення – двокрапкою.

Масив – послідовність значень, оточена квадратними дужками []. Елементи відділені комами.

Значення можуть бути наступних типів:

- рядок у подвійних лапках;
- число;
- логічні true або false;
- null;
- об'єкт;
- масив.

Ці структури можуть бути вкладеними одна в одну. Така структура даних зручна для обміну інформацією між компонентами системи.

5.2 Структура програмного комплексу планувальника місій

Програмний комплекс складається з наступних частин:

- модуля планування;
- модуля інтеграції;
- модуля візуалізації.

Обробка запитів, підготовка даних та обмін інформацією між програмою та зовнішніми сервісами відбувається у модулі інтеграції. Звідти підготовлена інформація про вхідні дані передається до модуля обчислення.

У модулі планування відбувається створення плану проведення місії з використанням розробленого алгоритму, результат роботи повертається у модуль керування.

Отриманий план обробляється і передається у модуль візуалізації, де формується мапа з вказанням планованих перельотів та текстовий опис місії. Вся інформація, необхідна для відображення, передається назад, у модуль керування, звідки виводиться користувачу чи системі, що розмістила задачу.

Модуль керування надає зовнішній інтерфейс, який дозволяє поставити завдання програмному комплексу та отримати план проведення місії.

Модуль обчислення отримує інформацію через стандартний потік вводу, відправляє результати через стандартний потік виводу. Модуль керування створює дочірній процес, у якому запущено екземпляр програми для обчислення, та обмінюється з нею інформацією для розв’язання задачі.

Модуль візуалізації надає інтерфейс для відправки матриці усіх точок задачі (рисунок 5.1) та визначених шляхів БПЛА, визначення характеристик БПЛА (рисунок 5.2), відображення та експорту плану місії, керування параметрами обчислення (рисунки 5.3-5.9).

Точки

Номер	Тип	Широта	Довгота
0	Ціль	50.19589993965801	30.048396028016914
1	Ціль	50.247301488502686	30.06315611746451
2	Ціль	50.26925107079161	30.124599280513724
3	Ціль	50.23764046824102	30.210413754046176
4	Ціль	50.17457621905148	30.224830585599626
5	Ціль	50.13608150547098	30.14210543311435
6	База	50.215236935107626	30.132150954184592
7	Ціль	50.149063186315715	30.074826885864923
8	Ціль	50.21589601277037	30.102630775289406
9	Ціль	50.235444508725195	30.13421050154937

Рисунок 5.1. Елемент інтерфейсу з відображенням списку цілей

БПЛА

Номер	Назва	Швидкість	Запас
1	HEE Wing F-01	100 км/год	130 км
2	HEE Wing F-02	100 км/год	130 км
3	HEE Wing F-02	100 км/год	130 км

Рисунок 5.2. Елемент інтерфейсу з відображенням списку БПЛА

```
class TargetState(Point):
    def __init__(self, target):
        self.position = target.position
        self.visited = False
        self.name = target.name
        super().__init__(*target.position)

    def __repr__(self):
        return '{} Target at {}:{}'.format(
            'Visited' if self.visited else '', *self.position)
```

Рисунок 5.3. Код класу, що відображає стан цілі

```
class Point:
    def __init__(self, x, y, free=False):
        self.x = x
        self.y = y
        self.free = free

    @lru_cache(None)
    def __sub__(self, other):
        if self.free:
            if isinstance(other, Depot):
                return 0
            else:
                return float('inf')
        if other.free:
            return float('inf')
        # return abs(self.x - other.x) + abs(self.y - other.y)
        return geodesic((self.x, self.y), (other.x, other.y)).km

    def __getitem__(self, key):
        if key == 0:
            return self.x
        elif key == 1:
            return self.y
        raise IndexError()

    def __eq__(self, other):
        return self.x == other.x and self.y == other.y

    def __hash__(self):
        return hash(self.x) + hash(self.y)
```

Рисунок 5.4. Код класу, що відображає стан точки на карті

```
class Vehicle:
    def __init__(self, dist, speed, color='c', name=''):
        self.speed = speed
        self.dist = dist
        self.color = color
        self.name = name

    @staticmethod
    def fabric(vehicles):
        colors = ['#aaaa00', '#aa00aa', '#00aa00']
        for i, vehicle in enumerate(vehicles):
            if isinstance(vehicle, str):
                dist, speed, name = vehicle.strip().split()
                yield Vehicle(float(dist), float(speed), name=name,
            elif isinstance(vehicle, dict):
                yield Vehicle(
                    vehicle['dist'], vehicle['speed'], name=vehicle.
            else:
                raise NotImplementedError

    def create_state(self, depots: [Depot]):
        return VehicleState(self, Point(0, 0, free=True), depots)

    def __repr__(self):
        return 'Vehicle {}'.format(self.name)
```

Рисунок 5.5. Код класу, що відображає стан агенту (БПЛА, дрон)

```python
class Greedy(Alg):
    """Greedy algorithm, stores all parameters and states while solving"""
    def __init__(self, data: Data, *args, **kwargs):
        super().__init__(data, *args, **kwargs)

    def get_score(self, vehicles=None):
        """Calculate a score of current state

        :return integer score"""
        score = 0
        for vehicle in (vehicles or self.vehicles):
            if vehicle.points:
                points = [p for p in vehicle.points]
                score += sum([p - n for p, n in zip(points, points[1:])])

        return score

    def _get_available_options(self, vehicle: VehicleState):
        """Select options available for the vehicle

        :param vehicle: current vehicle with its state
        :return AvailableOptions list"""
        return vehicle.distance_to_list([target for target in self.targets
                                         if not target.visited]) or []
```

Рисунок 5.6. Код оцінки плану місії жадібного алгоритму для команди БПЛА

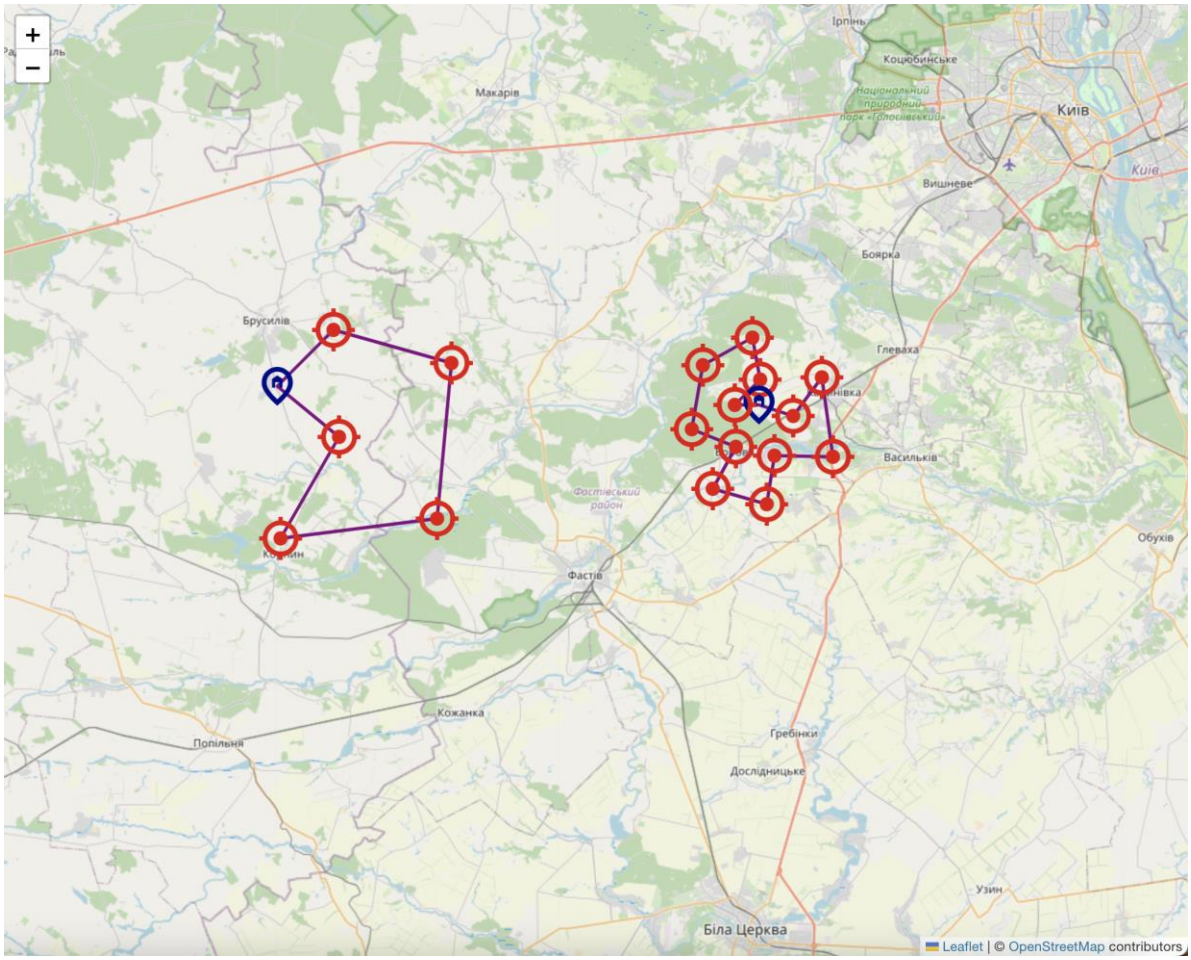

Рисунок 5.7. Приклад плану місії команди БПЛА з двома базами та двома агентами

p	a	b	Довжина	Час обрахунку	К-ть ітерацій	К-ть лок. пошуків	Далекозорий	Відхилення
0.1	0.1	0.5	172.3226511909352	2.00 с	7	19	ні	-29.86%
0.1	0.1	4	124.2486431058725	2.00 с	150	22	ні	-2.72%
0.1	0.1	7	133.1660567930686	2.00 с	150	21	ні	-9.23%
0.1	0.4	0.5	175.23453380129678	2.00 с	7	19	ні	-31.02%
0.1	0.4	4	120.87066406860546	2.00 с	142	22	ні	-
0.1	0.4	7	133.1660567930686	2.00 с	13	20	ні	-9.23%
0.1	0.8	0.5	140.9737478842774	2.00 с	14	19	ні	-14.26%
0.1	0.8	4	123.10519940648116	2.00 с	386	22	ні	-1.82%
0.1	0.8	7	133.1660567930686	2.00 с	55	22	ні	-9.23%
0.4	0.1	0.5	160.9850322973087	2.00 с	21	20	ні	-24.92%
0.4	0.1	4	134.8226456253305	2.00 с	55	21	ні	-10.35%
0.4	0.1	7	133.1660567930686	2.00 с	150	22	ні	-9.23%
0.4	0.4	0.5	169.82207686809278	2.00 с	7	20	ні	-28.83%
0.4	0.4	4	120.87066406860546	2.02 с	142	19	ні	-
0.4	0.4	7	133.1660567930686	2.00 с	13	16	ні	-9.23%
0.4	0.8	0.5	170.1119628436558	2.00 с	84	22	ні	-28.95%
0.4	0.8	4	134.9127289125231	2.00 с	187	23	ні	-10.41%
0.4	0.8	7	125.94983871881627	2.01 с	187	21	ні	-4.03%
0.8	0.1	0.5	160.9850322973087	2.00 с	21	18	ні	-24.92%
0.8	0.1	4	122.89318435400068	2.00 с	142	20	ні	-1.65%
0.8	0.1	7	133.1660567930686	2.00 с	254	22	ні	-9.23%
0.8	0.4	0.5	169.82207686809278	2.00 с	7	21	ні	-28.83%
0.8	0.4	4	132.50154941299604	2.00 с	111	23	ні	-8.78%
0.8	0.4	7	133.1660567930686	2.00 с	13	17	ні	-9.23%
0.8	0.8	0.5	166.00542884275512	2.00 с	27	19	ні	-27.19%

Рисунок 5.8. Екран аналізу результатів автоматичного вибору параметрів для запуску ОМК

Рисунок 5.9. Екран візуалізація плану місії команди БПЛА з трьома базами

5.3 Інтеграція обчислювального блоку з системами керування БПЛА

Розроблені алгоритми планування місії для БПЛА реалізовано у вигляді обчислювального блоку, що може бути інтегрований з популярними системами управління БПЛА, такими як UgCS, Mission Planner та APM Planner, використовуючи метод обміну файлами у сумісних форматах. Інтеграція обчислювального блоку з системами управління БПЛА дозволяє ефективно використовувати розроблені алгоритми планування місій у реальних умовах. Процес вимагає уважності на етапах підготовки даних, імпорту файлів та перевірки параметрів місії. За допомогою сумісних форматів файлів, забезпечується гнучкість та сумісність із різними платформами управління БПЛА.

Детальний опис процесу інтеграції представлений нижче.

5.3.1 UgCS (Universal Ground Control Software)

Планування місії виконується обчислювальним блоком, який генерує файли місії у форматі, сумісному з UgCS. Файли місії можуть бути представлені у форматі XML або JSON, який підтримується UgCS.

Для імпорту даних у UgCS достатньо вибрати опцію імпорту місії та вибрати файл місії, створений обчислювальним блоком. Після імпорту файлу, необхідно перевірити маршрут, точки та параметри місії в UgCS, щоб переконатися, що вони були імпортовані коректно.

Після імпорту можна налаштувати БПЛА, скоригувати маршрут (за потреби), вибрати опцію запуску місії в UgCS та слідкувати за ходом її виконання.

5.3.2 Mission Planner

Обчислювальний блок може генерувати файл місії у форматі, сумісному з Mission Planner (зазвичай у форматі .waypoints). За потреби, використовувати додаткові інструменти конвертації для перетворення формату файлу.

Імпорт даних у Mission Planner:

Для імпорту плану місії потрібно відкрити Mission Planner, вибрати опцію "Flight Plan", далі "Load WP file" та вибрати файл місії. Після перевірки та коригування плану, налаштування БПЛА та підготовки до польотів вибрати опцію запуску місії.

5.3.3 APM Planner

Процес інтеграції та використання APM Planner є практично аналогічним до процесу використання Mission Planner, оскільки обидві ці системи розроблені на базі того самого програмного забезпечення.

5.4 Висновки до розділу

Створено обчислювальний блок для реалізації розроблених алгоритмів планування місій гібридних систем та команд БПЛА. У блоці застосовано сучасні технології та архітектурні рішення для ефективної обробки даних. Розроблений та використаний формат обміну даними між компонентами

системи дозволяє легко описувати цілі, бази, БПЛА та їх характеристики у зручному для обробки вигляді.

Обчислювальний блок складається з модуля планування, де виконуються розрахунки, модуля інтеграції для обміну даними та модуля візуалізації результатів. Така архітектура забезпечує гнучкість та можливість розширення функціоналу. Передбачено можливість інтеграції розробленого блоку з популярними системами управління БПЛА, такими як UgCS, Mission Planner, APM Planner. Це дозволяє використовувати розроблені алгоритми на практиці при створенні плану місій, що виконуються транспортом та БПЛА у реальних умовах.

Загалом розроблений обчислювальний блок є гнучким та ефективним інструментом для реалізації та запуску алгоритмів планування місій. Він може бути інтегрований у різні системи та легко розширюватися для розв’язування нових задач.

РОЗДІЛ 6. РЕЗУЛЬТАТИ ЧИСЕЛЬНИХ ЕКСПЕРИМЕНТІВ

6.1 Результати дослідження алгоритмів оптимізації плану місій з гібридними системами

Для оцінки застосовності пропонованого підходу до планування місії гібридної системи «Дрон+ТЗ» проведено обчислювальний експеримент із розв'язування низки задач, сформованих на базі використання даних по задачам комівояжера із відомої бібліотеки TSPLIB, деякі із точок в яких вибиралися як точки зльоту та посадки на маршруті ТЗ.

Три з поданих задач було сформовано на реальних геоданих, а чотири – шляхом використання відомих задач із бібліотеки TSPLIB:

- задача 1 з 48 цілями та 4 базами, топологічно базується на задачі berlin52 з TSPLIB;
- задача 2 з 15 цілями та 5 базами;
- задача 3 з 24 цілями та 5 базами;
- задача 4 з 19 цілями та 3 базами;
- задача 5 з 12 цілями та 3 базами;
- задача 6 з 11 цілями та 3 базами, топологічно базується на задачі burma14 з TSPLIB;
- задача 7 з 15 цілями та 4 базами;
- задача 8 з 18 цілями та 4 базами;
- задача 9 з 36 цілями та 6 базами, топологічно базується на задачі danzig42 з TSPLIB;
- задача 10 з 19 цілями та 3 базами, топологічно базується на задачі ulysses22 з TSPLIB.

Для кожної задачі було виконано автоматичне налаштування параметрів мурашиного алгоритму шляхом формування спеціальної задачі оптимізації. З кожним набором параметрів виконано 3 запуски з різними ініціалізаторами генератора псевдовипадкових чисел. Час роботи алгоритму в усіх запусках для підбору параметрів обмежено 20 с.

Обчислення проводилися на ПК з такими параметрами:

- Macbook Pro 16-inch 2019;
- Processor: 2,6 GHz 6-Core Intel Core i7;
  - Graphic: AMD Radeon Pro 5300M 4 GB;
  - Intel UHD Graphics 630 1536 MB;
- RAM: 16 GB 2667 MHz DDR4;
- Mac OS: 13.2.1 (22D68).

Параметри алгоритму ОМК ρ, α та β визначалися як такі, що відповідають найкращому розв'язку, отриманому на етапі автоматичного налаштування параметрів. Для обчислювального експерименту запас ходу БПЛА встановлено як 300 км. Результати експерименту наведено у табл. 6.1.

Таблиця 6.1. Результати експерименту

Номер задачі	n	b	S_{rec}, км	S_{loc}, км
1	52	4	201.15	208.68
2	20	5	308.41	341.86
3	29	5	209.99	244.96
4	22	3	322.93	326.63
5	15	3	321.24	321.24
6	14	3	91.03	104.65
7	19	4	176.37	196.49
8	22	4	135.88	149.03
9	42	6	156.84	196.92
10	22	3	476.31	486.48

Тут S_{rec} – сумарна довжина перельотів кожного БПЛА, що відповідає найкращому знайденому плану місії, S_{loc} – сумарна довжина перельотів кожного БПЛА, що відповідає плану місії, отриманого за допомогою комбінації ДЛП та жадібного алгоритму, n – розмірність задачі (кількість цілей та баз), b – кількість точок вильоту та посадки (баз).

Поданий на рис. 6.1 план відповідає задачі 2.

За обмеження ресурсу БПЛА у 300 км для відвідування усіх цілей достатньо використання двох баз та одного циклу на кожній. Для експерименту зменшимо ресурс до 170 км – результат візуалізовано на рисунку 6.2.

У цьому випадку БПЛА робить чотири підцикли з двох баз, оскільки обмеження запасу ходу не дозволяє відвідати всі цілі у рамках двох підциклів, як на рис. 6.1. Загальна довжина перельотів при цьому зростає до 397.56 км.

Рисунок 6.1. Візуалізація отриманого плану місії по задачі 2

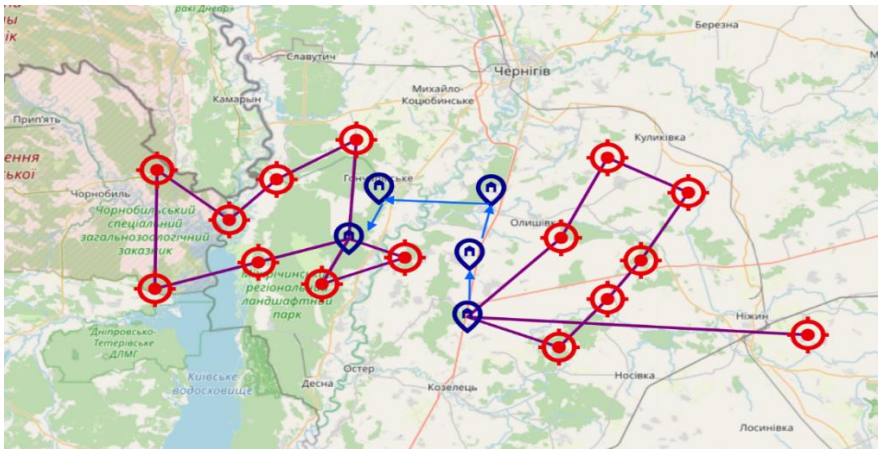

Рисунок 6.2. Візуалізація отриманого плану місії зі зменшеним ресурсом

6.2 Результати дослідження алгоритмів оптимізації плану командних місій

Для оцінки застосовності пропонованого підходу до планування у реальному часі проведено обчислювальний експеримент із розв'язування низки задач оптимізації маршрутів команди БПЛА, сформованих на базі використання даних по задачам комівояжера із відомої бібліотеки TSPLIB [13], деякі із точок в яких вибиралися як бази.

Три задачі було сформовано на реальних геоданих, а чотири – шляхом використання відомих задач із бібліотеки TSPLIB:

- Задача 1 з 48 цілями та 4 базами, топологічно базується на задачі berlin52 з TSPLIB;
- Задача 2 з 15 цілями та 5 базами;
- Задача 3 з 24 цілями та 5 базами;
- Задача 4 з 19 цілями та 3 базами;
- Задача 5 з 12 цілями та 3 базами;
- Задача 6 з 11 цілями та 3 базами, топологічно базується на задачі burma14 з TSPLIB;
- Задача 7 з 15 цілями та 4 базами;
- Задача 8 з 18 цілями та 4 базами;
- Задача 9 з 36 цілями та 6 базами, топологічно базується на задачі danzig42 з TSPLIB;
- Задача 10 з 19 цілями та 3 базами, топологічно базується на задачі ulysses22 з TSPLIB.

Для кожної задачі було виконано попередній підбір параметрів мурашиного алгоритму за допомогою пришвидшених запусків з меншою кількістю ітерацій. З кожним набором параметрів виконано 3 запуски з різними ініціалізаторами генератора псевдовипадкових чисел. Час роботи алгоритму в усіх запусках для підбору параметрів обмежено 20 секундами.

Обчислення проводилися на ПК з такими параметрами:

- Macbook Pro 16-inch 2019;
- Processor: 2,6 GHz 6-Core Intel Core i7;
- Graphic:

  - AMD Radeon Pro 5300M 4 GB;
  - Intel UHD Graphics 630 1536 MB;
- RAM: 16 GB 2667 MHz DDR4;
- Mac OS: 13.2.1 (22D68).

Параметри алгоритму ОМК ρ, α та β визначалися як ті, що відповідають найкращому розв’язку, отриманому на етапі налаштування параметрів.

Для експерименту розрахунковий запас ходу кожного БПЛА складає 600 км. Кількість доступних БПЛА – 2.

Кожна задача розв'язувалася шляхом трьох запусків запропонованого алгоритму з різними ініціалізаторами генератора псевдовипадкових чисел. Основні результати проведеного експерименту подано у табл. 6.2.

Таблиця 6.2. Результати алгоритмів оптимізації плану командних місій

Номер задачі	***n***	***b***	S_{rec}**, км**	S_{loc}**, км**
1	52	4	186,33	193.86
2	20	5	296,89	316.66
3	29	5	202,89	223.58
4	22	3	318,65	326.63
5	15	3	313,64	315.55
6	14	3	102,75	102.99
7	19	4	176,37	209.36
8	22	4	129,76	137.84
9	42	6	154,58	182.29
10	22	3	460,86	469.48

Тут S_{rec} – сумарна довжина перельотів кожного БПЛА, що відповідає найкращому знайденому плану місії, S_{loc} – сумарна довжина перельотів кожного БПЛА, що відповідає плану місії, отриманого за допомогою ДЛП, n – розмірність задачі (кількість цілей та баз), b – кількість баз.

Рисунок 6.3. Візуалізація отриманого плану місій для задачі 5 за допомогою модифікованого ОМК.

На рис. 6.3 зображено план місії, отриманий для задачі 5 (15 точок, 3 бази). Пронумеруємо бази від 1 до 3 відповідно до підписів на рисунку. Тоді план місії можна інтерпретувати наступним чином:

- БПЛА №1 злетів з бази №1, відвідав 7 цілей, оновив запас ходу на базі №2, відвідав 5 цілей, завершив політ на базі №3;
- БПЛА №2 не було задіяно.

Одна з альтернативних інтерпретацій:

- БПЛА №1 злетів з бази №1, відвідав 7 цілей, завершив політ на базі №2;
- БПЛА №2 злетів з бази №2, відвідав 5 цілей, завершив політ на базі №3.

Оскільки для обох випадків загальна відстань є однаковою, вибір конкретної інтерпретації не впливає на цільову функцію, проте у випадку оптимізації місії за часом це може впливати на загальний час через можливість паралельної роботи БПЛА. Аналогічно, у даному випадку на цільову функцію не впливає зміна напрямку руху БПЛА.

Для порівняння, розглянемо план, отриманий за допомогою жадібного алгоритму та локального пошуку.

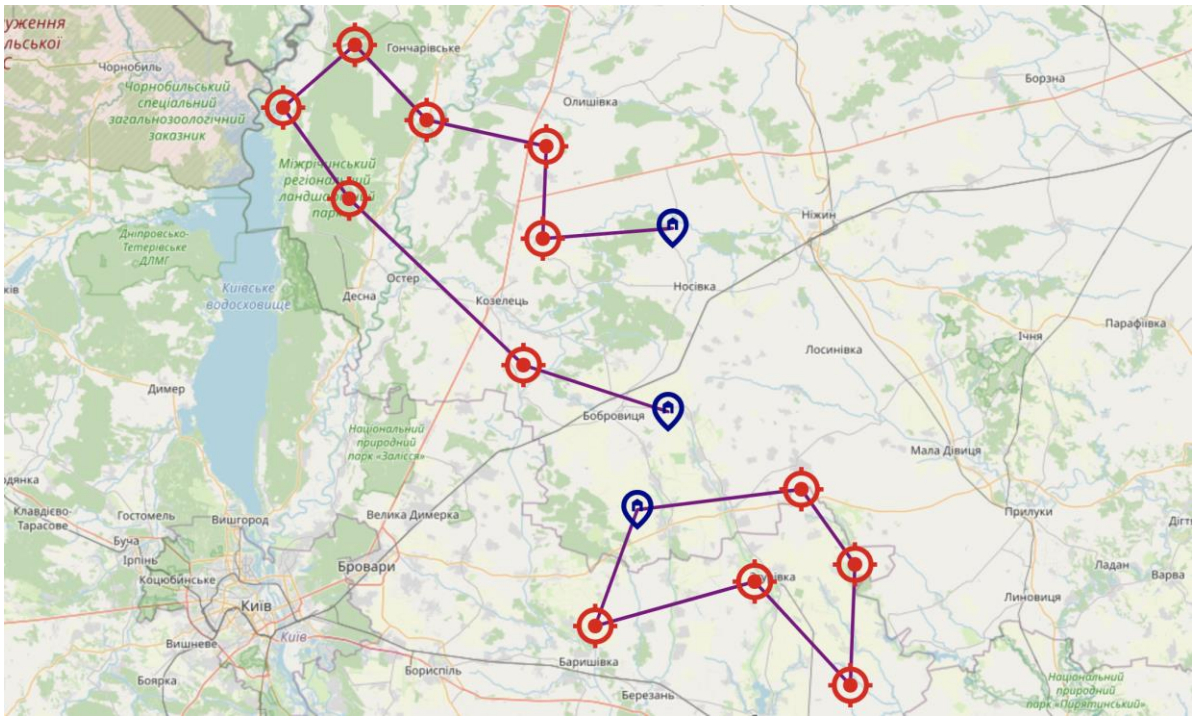

Рисунок 6.4. Візуалізація отриманого плану місій для задачі 5 за допомогою модифікованого ДЛП.

Слід зазначити, що на задачах невеликої розмірності різниця між результатами модифікованих ДЛП та ОМК виявилася значно меншою, ніж на задачах більшої розмірності зі складними підмаршрутами.

Розглянемо також плани, побудовані для задачі 3.

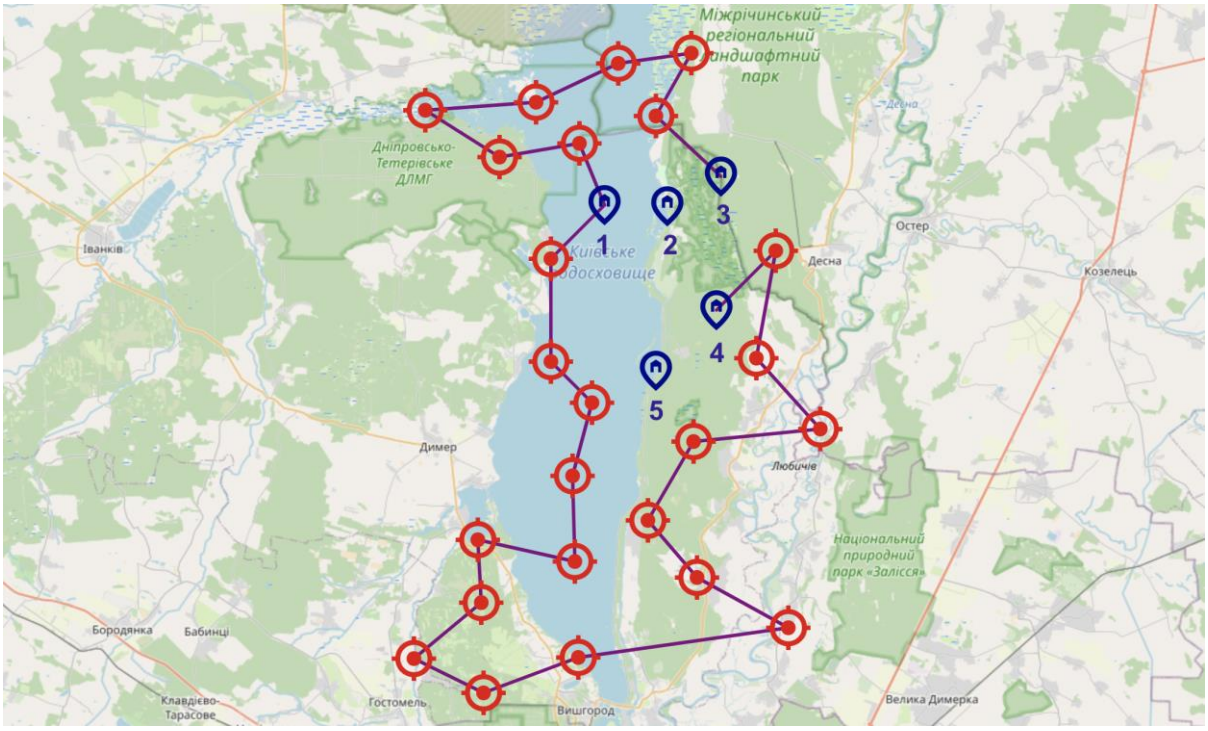

Рисунок 6.5. Візуалізація отриманого плану місій для задачі 3, отриманого за допомогою модифікованого ОМК.

На рис. 6.5 зображено план місії, отриманий для задачі 3 (29 точок, 5 баз). Пронумеруємо бази від 1 до 5 відповідно до підписів на рисунку. Тоді план місії можна інтерпретувати наступним чином:

- БПЛА №1 злетів з бази №1, відвідав 7 цілей, завершив політ на базі №3;
- БПЛА №2 злетів з бази №1, відвідав 17 цілей, завершив політ на базі №4;
- Бази №2 та №5 залишилися незадіяними.

У табл. 6.3 наведено результати запусків алгоритму, отримані з різними комбінаціями параметрів ρ, α та β, часове обмеження – 5 секунд на кожен запуск алгоритму. Відхилення вказано у порівнянні із найкращим знайденим результатом (у даному випадку він відповідає комбінаціям ρ=0.8, α=0.4, β=4 та ρ=0.4, α=0.8, β=4).

Таблиця 6.3. Результати запусків алгоритму

ρ	**α**	**β**	**Довжина, км**	**Відхилення, %**
0.1	0.1	0.5	285	28,93
0.1	0.1	4	226,14	10,28
0.1	0.1	7	215,78	5,97
0.1	0.4	0.5	278,26	27,08
0.1	0.4	4	210,96	3,82
0.1	0.4	7	210,23	3,49
0.1	0.8	0.5	281,48	27,92
0.1	0.8	4	204,06	0,57
0.1	0.8	7	206,67	1,83
0.4	0.1	0.5	265,68	23,63
0.4	0.1	4	217,83	6,86
0.4	0.1	7	211,01	3,85
0.4	0.4	0.5	299,25	32,20
0.4	0.4	4	206,64	1,81
0.4	0.4	7	205,27	1,16
0.4	0.8	0.5	271,37	25,23
0.4	0.8	4	202,89	0
0.4	0.8	7	206,68	1,83
0.8	0.1	0.5	276,07	26,51
0.8	0.1	4	212,93	4,71
0.8	0.1	7	205,27	1,16
0.8	0.4	0.5	250,11	18,88
0.8	0.4	4	202,89	0
0.8	0.4	7	206,68	1,83
0.8	0.8	0.5	266,83	23,96
0.8	0.8	4	209,49	3,15
0.8	0.8	7	208,83	2,84

Відхилення обраховується як $(1 - S_{rec} / S) \cdot 100$, де S_{rec} – кращий отриманий результат, S – результат для відповідної комбінації параметрів. Обрана комбінація співпадає з комбінацією, вибраною під час фази автоматичного налаштування алгоритму. При запуску алгоритму зі збільшеним обмеженням часу окремі комбінації також призводять до отримання найкращого відомого розв’язку. Наприклад, при запуску алгоритму з обмеженням у 30 секунд та комбінаціями ρ=0.4, α=0.4, β=7 та ρ=0.8, α=0.4 та β=4, довжина побудованого плану місії складає 202,89 км.

Рис. 6.6 ілюструє динаміку пошуку кращого результату у залежності від часу запуску для двох груп параметрів, вказаних вище. Для першої комбінації

найкраще відоме значення отримується на 27 секунді роботи, в той час як для другої – через 1.2 секунди після запуску.

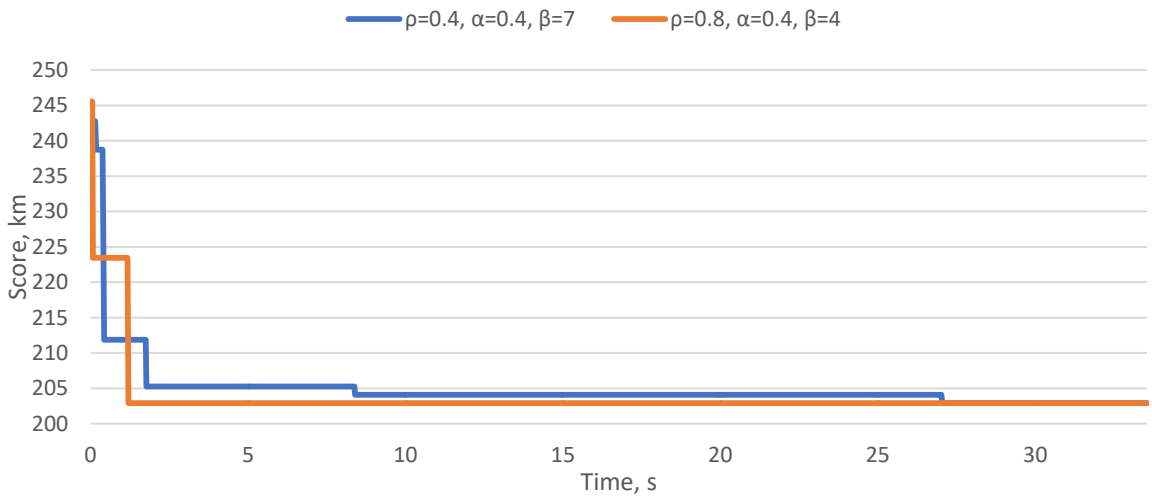

Рисунок 6.6. Візуалізація покращення результату в залежності від часу роботи алгоритму.

Для порівняння наводиться результат роботи модифікованого ДЛП для задачі 3.

Рисунок 6.7. Візуалізація плану місій для задачі 3, отриманого за допомогою модифікованого ДЛП.

Довжина найкращого маршруту, отриманого за допомогою ДЛП, дорівнює 223.58 км проти 202.89 для ОМК. Слід підкреслити, що запуски ДЛП також є частиною модифікованого ОМК, проте, кращі результати забезпечуються варіативністю початкових розв’язків.

6.3 Аналіз результатів чисельних експериментів та висновки

Розроблені алгоритми було апробовано на спеціально розроблених задачах з багатьма базами і строгими обмеженнями. Частина поданих задач базується на відомих та розповсюджених задачах комівояжера.

У цьому розділі представлено результати чисельних експериментів для оцінки ефективності розроблених алгоритмів оптимізації планів місій гібридних систем та команд БПЛА.

Для гібридних систем було сформовано 10 тестових задач на основі геоданих та окремих задач з TSPLIB. Проведено порівняння розробленого алгоритму оптимізації мурашиною колонією з поєднанням жадібного алгоритму та локального пошуку. Результати показали перевагу розробленого алгоритму ОМК, зокрема для задач великої розмірності.

Для командних місій безпілотників також сформовано 10 тестових задач на основі геоданих та окремих задач з TSPLIB та порівняно модифіковані алгоритми ОМК і детермінованого локального пошуку. Результати продемонстрували кращу ефективність запропонованого ОМК. Проаналізовано вплив параметрів та часу роботи на результати.

Отже, експерименти показали переваги розроблених алгоритмів для задач планування місій як гібридних систем, так і команд БПЛА. Це дає підґрунтя для подальшого вдосконалення та практичного застосування поданого підходу.

РОЗДІЛ 7. ВИСНОВКИ

1. Формалізовано нові задачі оптимізації рішень при плануванні місій з обстеження чи обслуговування заданих цілей окремими БПЛА, їх командами та гібридними системами "Дрон + ТЗ".

2. Запропоновано підхід до розв’язування складних задач комбінаторної оптимізації, що виникають, на основі використання ройового інтелекту, який дозволив агрегувати в єдину оптимізаційну задачу етапи розбиття цілей по базам, вибору задіяних баз та оптимізації маршрутів БПЛА з урахуванням обмеженості польотного ресурсу.

3. Розроблено, реалізовано програмно та досліджено алгоритми жадібного типу, детермінованого локального пошуку та ОМК для розв'язування нових задач планування місій БПЛА задач з альтернативними та динамічними базами. Проведено обчислювальний експеримент, який показав переваги алгоритму ОМК з точки швидкодії та ефективності навіть для задач підвищеної розмірності.

4. Запропонована методологія формалізації та алгоритми розв’язування сформульованих задач планування місій БПЛА застосовані при створенні програмно-алгоритмічних засобів оптимізації рішень при плануванні місій команди БПЛА чи з використанням комплексу "Дрон + ТЗ", зокрема, в умовах неповної ситуаційної обізнаності.

5. На основі програмної реалізації Блоків оптимізації маршрутів за наявності альтернативних або стаціонарних баз чи гібридної системи «БПЛА+ТЗ» розроблено інтерфейс із поширеними міжнародними системами керування БПЛА MissionPlanner та UgCS.

6. Окремі компоненти розробленого програмного забезпечення передано ВАТ "Меридіан" ім. С.П. Корольова ДП "Укроборонпром" для впровадження.

7. Хоча у фокусі досліджень були БПЛА, розроблені алгоритмічні засоби можуть бути розвиненими для застосувань у випадках, коли транспортний засіб

– це авто, БТР, танк, корабель чи авіалайнер, а замість БпАК (чи сумісно з ними) використовуються інші дрони (наземні чи надводні РРС).

Пропонований підхід дозволив однією агрегованою математичною моделлю охопити декілька етапів планування місії гібридної системи "Дрон+ТЗ". Розроблена математична модель дозволяє охоплювати і проблему вибору напряму подальшого руху транспорту, що знаходиться у певному місці, у залежності від аналізу результатів обстеження заданих цілей, у яких можуть знаходитися об'єкти для обстеження чи обслуговування.

Для розв'язування сформульованої задачі комбінаторної оптимізації розроблено алгоритми жадібного типу, детермінованого локального пошуку та ОМК. Результати обчислювального експерименту продемонстрували перевагу алгоритму ОМК над комбінованим алгоритмом "жадібний алгоритм+детермінований локальний пошук".

Алгоритм ОМК дозволяє розв'язувати прикладні задачі підвищеної розмірності за прийнятний час – 20 с на ноутбуку із середніми показниками продуктивності.

Запропонований математичний апарат може бути розвинений у напрямі планування місії гібридних систем з декількома дронами, що проводить до постановок, близьких до задач з альтернативними базами [28, 32]. Зі стрімким розвитком логістики із залученням дронів виникають нові проблеми, які для свого вирішення вимагають врахування додаткових умов використання дронів. Розроблений математичний апарат дозволяє переходити до розгляду задач, у яких місця базування на маршруті транспорту не задаються, а визначаються у залежності від конфігурації цілей, як, наприклад, у [17].

Напрямом подальших досліджень може бути урахування у математичній моделі характеристик процесу розряджання акумуляторів, врахування заборонених зон для польоту та наявні на місцевості перешкоди, погодних умов (напряму вітру). Ці перешкоди можуть бути апроксимовані геометричними фігурами, тому для розв'язування можуть використовуватися гібридні алгоритми із залученням методів Евклідової оптимізації [38].

Перспективним є покращення мурашиного алгоритму за рахунок використання диверсифікованих алгоритмів пошуку розв'язків [19], а також паралельної реалізації острівної моделі алгоритму ОМК [15].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Thibbotuwawa A., Bocewicz G., Nielsen P., Banaszak, Z. (2020). Unmanned aerial vehicle routing problems: a literature review. Applied sciences, 10(13), 4504. https://doi.org/10.1111/itor.12783
2. Macrina G., Di Puglia Pugliese L., Guerriero F, Laporte G. (2020). Drone-aided routing: A literature re-view. Transportation Research Part C. Volume 120, November 2020, 102762 https://doi.org/10.1016/j.trc.2020.102762
3. Huang, S. H., Huang, Y. H., Blazquez, C. A., & Chen, C. Y. (2022). Solving the vehicle routing problem with drone for delivery services using an ant colony optimization algorithm. Advanced Engineering Infor-matics, 51, 101536. https://doi.org/10.1016/j.aei.2022.101536
4. Abubakar, A. I., Mollel, M. S., Onireti, O., Ozturk, M., Ahmad, I., Asad, S. M., ... & Imran, M. A. (2023). Coverage and throughput analysis of an energy efficient UAV base station positioning scheme. Computer Networks, Volume 232, August 2023, 109854. https://doi.org/10.1016/j.comnet.2023.109854
5. Liu, X., Jing, T., & Hou, L. (2023). An FW–GA Hybrid Algorithm Combined with Clustering for UAV Forest Fire Reconnaissance Task Assignment. Mathematics, 11(10), 2400. https://doi.org/10.3390/math11102400
6. Li, J.; Zhao, H.; Wang, H.; Gu, F.; Wei, J.; Yin, H.; Ren, B. (2020). Joint Optimization on Trajectory, Altitude, Velocity, and Link Scheduling for Minimum Mission Time in UAV-Aided Data Collection. IEEE Internet Things J. 2020, 7, 1464–1475. https://doi.org/10.1109/JIOT.2019.2955732
7. Li, J., Xiong, Y., & She, J. (2023). UAV path planning for target coverage task in dynamic environment. IEEE Internet of Things Journal. https://doi.org/10.1109/JIOT.2023.3277850
8. Kliushnikov I., Fesenko H., Kharchenko V., Illiashenko O., Morozova O. UAV fleet based accident moni-toring systems with automatic battery replacement systems: Algorithms for justifying composition and use planning. International

Journal of Safety and Security Engineering. 2021. Vol. 11, N 4. P. 319–328. https://doi.org/10.18280/ijsse.110404

9. Rakhmetullina, S., Zhomartkyzy, G., Krak, I., & Kamelova, A. (2023). Development of an Algorithm for Solving an Asymmetric Routing Problem Based on the Ant Colony Method. Cybernetics and Systems Analysis, 1-9. https://doi.org/10.1007/s10559-023-00588-w
10. Ahn N., Kim S. (2022). Optimal and heuristic algorithms for the multi-objective vehicle routing problem with drones for military surveillance operations. Journal of Industrial and Management Optimization, 18(3), 1651-1663. https://doi:10.3934/jimo.2021037
11. Golden, B., Wang, X., Wasil, E. (2023). The Evolution of the Vehicle Routing Problem – A Survey of VRP Research and Practice from 2005 to 2022. The Evolution of the Vehicle Routing Problem. Synthesis Lectures on Operations Research and Applications, Springer, Cham. 2023. P. 1-64. https://doi.org/10.1007/978-3-031-18716-2_1
12. Jeong, H. Y., & Lee, S. (2023). Drone routing problem with truck: Optimization and quantitative analysis. Expert Systems with Applications, 227, 120260. https://doi.org/10.1016/j.eswa.2023.120260
13. Горбулін В.П., Гуляницький Л.Ф., Сергієнко І.В. (2019). Постановки та математичні моделі проблем оптимізації маршрутів літальних апаратів із динамічними депо. Управляющие системы и машины, 1, 3-10. https://doi.org/10.15407/usim.2019.01.003
14. Horbulin, V.P., Hulianytskyi, L.F. & Sergienko, I.V. (2020). Optimization of UAV Team Routes in the Presence of Alternative and Dynamic Depots. Cybern Syst Anal 56, 2, 195–203. https://doi.org/10.1007/s10559-020-00235-8
15. Комплект з трьох БПЛА з вертикальним зльотом Бджола-20. prozorro.gov.ua. URL: https://prozorro.gov.ua/tender/UA-2023-06-15-015934-a (дата звернення: 03.01.2023).
16. ДК 021:2015: 34710000-7 Вертольоти, літаки, космічні та інші літальні апарати з двигуном (34711200-6 Безпілотні літальні апарати) (безпілотний

авіаційний комплекс з комбінованою системою керування). prozorro.gov.ua. URL: https://prozorro.gov.ua/tender/UA-2023-10-19-005700-a (дата звернення: 03.01.2023).

17. Poikonen, S., & Golden, B. (2020). The mothership and drone routing problem. INFORMS Journal on Computing, 32(2), 249-262. https://doi.org/10.1287/ijoc.2018.0879
18. Murray C.C., Raj R. (2020). The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transportation Research Part C: Emerging Technologies, 2020. Vol. 110. P. 368–398. https://doi.org/10.1016/j.trc.2019.11.003
19. Гуляницький Л.Ф., Рибальченко О.В. Оптимізація маршрутів при плануванні місій гібридних транспортних систем "Дрон+Транспортний засіб". Cybernetics and Computer Technologies. 2023. 3. С. 44–58. https://doi.org/10.34229/2707-451X.23.3.4
20. Гуляницький Л.Ф., Рибальченко О.В. Формалізація проблеми оптимізації місць базування та маршрутів групи БПЛА. Cybernetics and Computer Technologies. 2021. 4. С. 12–26. https://doi.org/10.34229/2707-451X.21.4.2
21. Гуляницький Л.Ф., Рибальченко О.В. Формалізація та розв'язування одного типу задач маршрутизації БПЛА. Теорія оптимальних рішень. 2018. 17. С. 107-114. http://dspace.nbuv.gov.ua/handle/123456789/144979
22. Hulianytskyi L. Rybalchenko O. Optimization of decisions when planning a UAV group mission with alternative depots. International Scientific Symposium «Intelligent Solutions», September 27-28, 2023, Kyiv, Ukraine. CEUR Workshop Proceedings. 2023. https://ceur-ws.org/Vol-3538/Paper_22.pdf
23. Kamil A. Alotaibi (2014). Unmanned Aerial Vehicle Routing In The Presence Of Threats. Arlington: The University Of Texas At Arlington. 25-121.
24. Murray, Chase & Raj, Ritwik. (2020). The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transportation Research Part C: Emerging Technologies. 110. 368-398. https://doi.org/10.1016/j.trc.2019.11.003

25. Murray C. C., Raj R. The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transportation Research Part C: Emerging Technologies. 2020. Vol. 110. P. 368-398. https://doi.org/10.1016/j.trc.2019.11.003
26. Wang, X., Poikonen, S., & Golden, B. (2017). The vehicle routing problem with drones: several worst-case results. Optimization Letters, 11, 679-697.
27. Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. Transportation research part B: methodological, 122, 350-364. https://doi.org/10.1016/j.trb.2019.03.005
28. Kuo, R. J., Lu, S. H., Lai, P. Y., & Mara, S. T. W. (2022). Vehicle routing problem with drones considering time windows. Expert Systems with Applications, 191, 116264. https://doi.org/10.1016/j.eswa.2021.116264
29. Jeong, H. Y., & Lee, S. (2021). Collaborative hybrid delivery system: Drone routing problem assisted by truck. In Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems: IFIP WG 5.7 International Conference, APMS 2021, Nantes, France, September 5–9, 2021, Proceedings, Part III (pp. 33-42). Springer Int. Publ.
30. Golden, B., Wang, X., & Wasil, E. (2023). The Evolution of the Vehicle Routing Problem – A Survey of VRP Research and Practice from 2005 to 2022. In The Evolution of the Vehicle Routing Problem: A Survey of VRP Research and Practice from 2005 to 2022 (pp. 1-64). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-18716-2
31. Gonzalez-R, P. L., Canca, D., Andrade-Pineda, J. L., Calle, M., & Leon-Blanco, J. M. (2020). Truck-drone team logistics: A heuristic approach to multi-drop route planning. Transportation Research Part C: Emerging Technologies, 114, 657-680. https://doi.org/10.1016/j.trc.2020.02.030
32. Zheng C. (2023). Review on Combined Vehicle Routing Problem of Drones and Vehicles. Journal of Innovation and Development, 2(1), 55–57. https://doi.org/10.54097/jid.v2i1.5420

33. Рибальченко О. В. Задача маршрутизації безпілотних літальних апаратів: Магістерська дисертація. Київ, 2018.
34. Yoo W., Yu E., Jung J. (2018). Drone delivery: Factors affecting the public's attitude and intention to adopt. Telematics and Informatics. V. 35, Issue 6, September 2018, Pages 1687-1700. https://doi.org/10.1016/j.tele.2018.04.014
35. Kyriakakis, N. A., Stamadianos, T., Marinaki, M., & Marinakis, Y. (2022). The electric vehicle routing problem with drones: An energy minimization approach for aerial deliveries. Cleaner Logistics and Supply Chain, 4, 100041. https://doi.org/10.1016/j.clscn.2022.100041
36. Kliushnikov I., Fesenko H., Kharchenko V., Illiashenko O., Morozova O. (2021). UAV fleet based accident monitoring systems with automatic battery replacement systems: Algorithms for justifying composition and use planning. Int. J. Saf. Secur. Eng, 11(4), 319-328. https://doi.org/10.18280/ijsse.110404
37. Yakovlev, S., Kartashov, O., Mumrienko, A. (2022). Formalization and Solution of the Maximum Area Coverage Problem Using Library Shapely for Territory Monitoring. Radioelectronic and Computer Systems, 2022(2), pp. 35–48. https://doi.org/10.32620/reks.2022.2.03
38. Gonzalez-R, P. L., Canca, D., Andrade-Pineda, J. L., Calle, M., & Leon-Blanco, J. M. (2020). Truck-drone team logistics: A heuristic approach to multi-drop route planning. Transportation Research Part C: Emerging Technologies, 114, 657-680. https://doi.org/10.1016/j.trc.2020.02.030
39. Poikonen, S., & Golden, B. (2020). Multi-visit drone routing problem. Computers & Operations Research, 113, 104802. https://doi.org/10.1016/j.cor.2019.104802
40. Tamke, F., & Buscher, U. (2023). The vehicle routing problem with drones and drone speed selection. Computers & Operations Research, 152, 106112. https://doi.org/10.1016/j.cor.2022.106112
41. Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. Transportation research part B: methodological, 122, 350-364. https://doi.org/10.1016/j.trb.2019.03.005

42. Ramos, T. R. P., Gomes, M. I., & Póvoa, A. P. B. (2020). Multi-depot vehicle routing problem: a comparative study of alternative formulations. International Journal of Logistics Research and Applications, 23(2), 103-120. https://doi.org/10.1080/13675567.2019.1630374
43. Jeong, H. Y., & Lee, S. (2021). Collaborative hybrid delivery system: Drone routing problem assisted by truck. In Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems: IFIP WG 5.7 International Conference, APMS 2021, Nantes, France, September 5–9, 2021, Proceedings, Part III (pp. 33-42). Springer Int. Publ.
44. Murray C.C., Chu A.G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. Transportation Research Part C: Emerging Technologies. Volume 54, May 2015, Pages 86-109. https://doi.org/10.1016/j.trc.2015.03.005
45. Gonzalez-R, P. L., Canca, D., Andrade-Pineda, J. L., Calle, M., & Leon-Blanco, J. M. (2020). Truck-drone team logistics: A heuristic approach to multi-drop route planning. Transportation Research Part C: Emerging Technologies, 114, 657-680. https://doi.org/10.1016/j.trc.2020.02.030
46. Zhang S., Liu S., Xu W., Wang W. (2022). A novel multi-objective optimization model for the vehicle routing problem with drone delivery and dynamic flight endurance. Computers & Industrial Engineering, 173, 108679. https://doi.org/10.1016/j.cie.2022.108679
47. Yurek, E. E., & Ozmutlu, H. C. (2021). Traveling salesman problem with drone under recharging policy. Com-puter Communications, 179, 35-49. https://doi.org/10.1016/j.comcom.2021.07.013
48. Meng, S., Guo, X., Li, D., & Liu, G. (2023). The multi-visit drone routing problem for pickup and delivery services. Transportation Research Part E: Logistics and Transportation Review, 169, 102990. https://doi.org/10.1016/j.tre.2022.102990

49. Poikonen, S., & Golden, B. (2020). The mothership and drone routing problem. INFORMS Journal on Compu-ting, 32(2), 249-262. https://doi.org/10.1287/ijoc.2018.0879
50. Murray C.C., Raj R. (2020). The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transportation Research Part C: Emerging Technologies, 2020. Vol. 110. P. 368–398. https://doi.org/10.1016/j.trc.2019.11.003
51. Leon-Blanco, J. M., Gonzalez-R, P. L., Andrade-Pineda, J. L., Canca, D., & Calle, M. (2022). A multi-agent approach to the truck multi-drone routing problem. Expert Systems with Applications, 195, 116604. https://doi.org/10.1016/j.eswa.2022.116604
52. Poikonen, S., Golden, B. (2020). Poikonen, S., & Golden, B. (2020). Multi-visit drone routing problem. Computers & Operations Research, 113, 104802. https://doi.org/10.1016/j.cor.2019.104802
53. Rojas Viloria, D., Solano-Charris, E. L., Muñoz-Villamizar, A., & Montoya-Torres, J. R. (2021). Unmanned aerial vehicles/drones in vehicle routing problems: a literature review. Int. Transactions in Operational Research, 28(4), 1626-1657. https://doi.org/10.1111/itor.12783
54. Luo Z., Poon M., Zhang Z., Liu Z., Lim A. (2021). The multi-visit traveling salesman problem with multi-drones. Transportation Research Part C: Emerging Technologies, 128 (April), Article 103172. https://doi.org/10.1016/j.trc.2021.103172
55. Morandi, N., Leus, R., Matuschke, J., & Yaman, H. (2023). The Traveling Salesman Problem with Drones: The Benefits of Retraversing the Arcs. Transportation Science, Vol. 57, No. 5. P. 1340–1358. https://doi.org/10.1287/trsc.2022.0230
56. Tamke, F., & Buscher, U. (2023). The vehicle routing problem with drones and drone speed selection. Comput-ers & Operations Research, 152, 106112. https://doi.org/10.1016/j.cor.2022.106112

57. Cengiz, E., Yilmaz, C., Kahraman, H., Suiçmez, Ç. (2023). Effects of Variable UAV Speed on Optimization of Travelling Salesman Problem with Drone (TSP-D). In: Smart Applications with Advanced Machine Learning and Human-Centred Problem Design. ICAIAME 2021. Engineering Cyber-Physical Systems and Critical Infra-structures, vol 1. Springer, Cham. https://doi.org/10.1007/978-3-031-09753-9_23
58. Jeong, H.Y., Lee, S. (2021). Collaborative Hybrid Delivery System: Drone Routing Problem Assisted by Truck. In: Dolgui, A., Bernard, A., Lemoine, D., von Cieminski, G., Romero, D. (eds) Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems. APMS 2021. IFIP Advances in Information and Communication Technology, vol 632. Springer, Cham. https://doi.org/10.1007/978-3-030-85906-0_4
59. Jeong, H. Y., & Lee, S. (2023). Drone routing problem with truck: Optimization and quantitative analysis. Ex-pert Systems with Applications, 227, 120260. https://doi.org/10.1016/j.eswa.2023.120260
60. Horbulin, V.P., Hulianytskyi, L.F., Sergienko, I.V. (2020). Optimization of UAV Team Routes in the Presence of Alternative and Dynamic Depots. Cybernetics and Systems Analysis, 56(2). P. 195–203. https://doi.org/10.1007/s10559-020-00235-8
61. Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. Transportation research part B: meth-odological, 122, 350-364. https://doi.org/10.1016/j.trb.2019.03.005
62. Chung S.H., Sah B., Lee J. (2020). Optimization for Drone and Drone-truck Combined Operations: A Review of the State of the Art and Future Directions. Computers and Operations Research, Volume 123, November 2020, 105004. https://doi.org/10.1016/j.cor.2020.105004
63. Moshref-Javadi, M., Winkenbach, M. (2021). Applications and Research avenues for drone-based models in lo-gistics: A classification and review.

Expert Systems with Applications. Expert Systems with Applications, 177, 114854. https://doi.org/10.1016/j.eswa.2021.114854

64. Garey, M. R., & Johnson, D. S. (1979). Computers and intractability: A guide to the theory of NP-completeness. New York, USA: W.H. Freeman & Co Ltd.
65. Laporte, G. (1992). The vehicle routing problem: An overview of exact and approximate algorithms. European Journal of Operational Research, 59, 345–358.
66. Baldacci, R., Toth, P., & Vigo, D. (2010). Exact algorithms for routing problems under vehicle capacity constraints. Annals of Operations Research, 175, 213–245
67. Solomon, M. M., & Desrosiers, J. (1988). Time window constrained routing and scheduling problems. Transportation Science, 22(1), 1–13.

ДОДАТОК ГРАФІЧНИЙ МАТЕРІАЛ

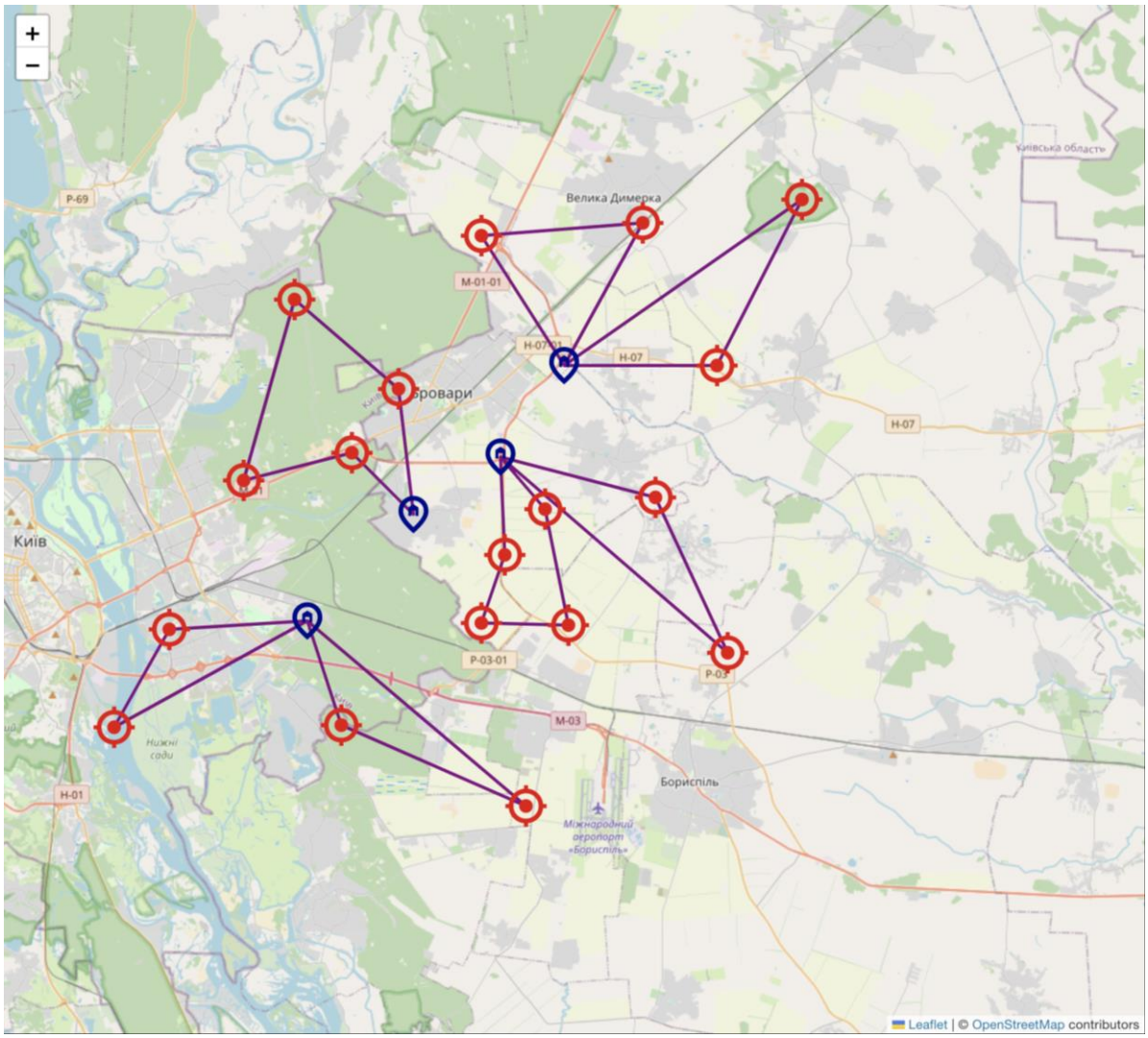

Рисунок 1. Приклад плану місії із багатьма петлями

Плани

БПЛА	Довжина	АРМ файл
HEE Wing F-01	58 км	
HEE Wing F-02	63 км	

Рисунок 2. Елемент інтерфейсу з експортом плану місії за БПЛА

```
QGC WPL 110
0	0	3	16	0	5	0	0	51.27909868682927	30.92955808999996	20	1
1	0	3	16	0	5	0	0	51.36835106051133	30.87189076378618	20	1
2	0	3	16	0	5	0	0	51.29627609493991	30.66877133751386	20	1
3	0	3	16	0	5	0	0	51.22236764552972	30.547944558780156	20	1
4	0	3	16	0	5	0	0	51.31344707827587	30.36395832752665	20	1
5	0	3	16	0	5	0	0	51.09662294502995	30.35846620122055	20	1
6	0	3	16	0	5	0	0	51.144894309328016	30.62208826391219	20	1
7	0	3	16	0	5	0	0	51.1052465308885	30.78401674288895	20	1
8	0	3	16	0	5	0	0	51.18795112740308	30.852668321714894	20	1
```

Рисунок 3. Формат даних для імпорту у Mission Planner

```python
from typing import List, Optional
from pydantic import BaseModel
from enum import Enum


class RoutingMode(Enum):
    round_trip = "round_trip"
    free_route = "free_route"


class Settings(BaseModel):
    routing_mode: RoutingMode = RoutingMode.free_route


class Data(BaseModel):
    depots: List[dict]
    targets: List[dict]
    vehicles: List[dict]
    settings: Optional[Settings]
```

Рисунок 4. Опис базових типів для вхідних даних задачі

```
let point_id_seq = 0

export class Point {
    constructor(x, y, type, id = null) {
        this.x = x
        this.y = y
        this.type = type
        this.id = id || point_id_seq++
    }

    static reset_seq() {
        point_id_seq = 0
    }
}
```

Рисунок 5. Клас, що описує точку на мапі у візуалізаторі

```
class FlightsReader(Reader):
    def process(self):
        with open(self.input_file, 'r') as f:
            header = Header(*[int(i) for i in f.readline().split()])
            depots = list(Depot.fabric(
                [next(f) for x in range(header.depots)]
            ))
            targets = list(Target.fabric(
                [next(f) for x in range(header.targets)]
            ))
            obstacles = list(Obstacle.fabric(
                [next(f) for x in range(header.obstacles)]
            ))
            vehicles = list(Vehicle.fabric(
                [next(f) for x in range(header.vehicles)]
            ))
            return Data(header, depots, targets, obstacles, vehicles)
```

Рисунок 6. Клас, що описує зчитувач вхідних умов задачі з файлу

```
def display(self, vehicles=None):
    g = Graph()

    g.add_points(self.targets + self.obstacles + self.depots)

    for v in (vehicles or self.vehicles):
        if not v.points:
            continue
        for p, n in zip(v.points, v.points[1:]):
            g.add_edge(Edge(
                self.points_transformer[p],
                self.points_transformer[n],
                v.vehicle.color))

    p = Plotter(g)
    p.plot_points()
    p.plot_edges()
    p.show()
```

Рисунок 7. Функція, що відображає проміжний план місії для налагоджування