ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ ТА МЕРЕЖ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «**ІНТЕЛЕКТУАЛЬНІ МЕТОДИ УНИКНЕННЯ ЗІТКНЕНЬ У ГРУПОВИХ ПОЛЬОТАХ БПЛА НА ОСНОВІ РЛС**»

на здобуття освітнього ступеня магістра
зі спеціальності 172 Електронні комунікації та радіотехніка
(шифр і назва спеціальності)
освітньо-професійної програми Телекомунікаційні системи та мережі
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ Микола БОВСУНІВСЬКИЙ
(підпис) Ім'я, ПРІЗВИЩЕ здобувача

Виконав: здобувач вищої освіти гр. ТСДМ-61
Микола БОВСУНІВСЬКИЙ
Ім'я, ПРІЗВИЩЕ
Керівник: к.т.н., доц. Катерина ДОМРАЧЕВА
Ім'я, ПРІЗВИЩЕ
Науковий ступень,
вчене звання
Резензент: _____
Ім'я, ПРІЗВИЩЕ
Науковий ступень
вчене звання

Київ 2025

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут Телекомунікацій

Кафедра Телекомунікаційних систем та мереж_____
Ступінь вищої освіти магістр_____
Спеціальність 172 Електронні комунікації та радіотехніка_____
Освітньо-професійна програма Телекомунікаційні системи та мережі

ЗАТВЕРДЖУЮ

Завідувач кафедри ТСМ

_____ Віктор ЗАЇКА

«____» _____2025 р.

_

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Бовсунівському Миколі Сергійовичу

1. Тема роботи: “ Інтелектуальні методи уникнення зіткнень у групових польотах БПЛА на основі РЛС”,

керівник роботи Домрачева Катерина Олексіївна, к.т.н., доцент,
затверджені наказом Державного університету інформаційно-комунікаційних технологій від 30.10. 2025 року № 467

2. Строк подання кваліфікаційної роботи: 12.12.2025 р.

3. Вихідні дані до кваліфікаційної роботи:
   1. Технічні характеристики сучасних FMCW-радарів міліметрового діапазону (76–81 ГГц).
   2. Набори даних (datasets) траєкторій БПЛА у щільних групах.
   3. Програмне середовище моделювання: ROS 2 Humble та Gazebo Ignition.
   4. Науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
   1. Аналіз сучасного стану технологій ройового управління БПЛА.
   2. Математичне моделювання динамічних перешкод та оцінка ризиків зіткнення.

3. Розробка гібридного алгоритму (Attention-Bi-GRU + ACO) для проактивної навігації.
4. Програмна реалізація та верифікація методу в умовах складного рельєфу та перешкод.

5. Перелік ілюстраційного матеріалу: презентація на 22 слайдах.

6. Дата видачі завдання 04.09.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Збір та аналіз актуальних джерел (2024-2025)	20.09.25	Викон.
2.	Розробка та навчання моделі Attention-Bi-GRU	15.10.25	Викон.
3.	Інтеграція прогнозу в алгоритм ACO	05.11.25	Викон.
4.	Проведення симуляцій в ROS 2 / Gazebo	20.11.25	Викон.
5.	Аналіз результатів та оформлення ПЗ	05.12.25	Викон.
6.	Подання роботи на кафедру та передзахист	12.12.25	Викон.
7.	Презентація	13.12.25	Викон.
8.	Збір та аналіз актуальних джерел (2024-2025)	20.09.25	Викон.
9.	Розробка та навчання моделі Attention-Bi-GRU	15.10.25	Викон.

Здобувач вищої освіти Микола БОВСУНІВСЬКИЙ

(підпис) (Ім'я,ПРІЗВИЩЕ)

Керівник кваліфікаційної роботи Катерина ДОМРАЧЕВА

(підпис) (Ім'я,ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 109 с., 37 рис., 4 табл., 44 джерела.

Мета роботи – підвищення надійності автономної навігації БПЛА шляхом впровадження інтелектуальної моделі прийняття рішень для уникнення зіткнень на основі даних РЛС.

Об'єкт дослідження – безпека автономних групових польотів БПЛА в умовах динамічних перешкод.

Предмет дослідження – методи та алгоритми уникнення зіткнень із використанням бортових радіолокаційних систем та штучного інтелекту.

Короткий зміст роботи: У кваліфікаційній магістерській роботі проведений аналіз децентралізованих навігаційних архітектур та технологій обробки сигналів FMCW-радарів. Розроблено гібридну методологію, що поєднує нейронну мережу Attention-Bi-GRU для прогнозування траєкторій та мурашиний алгоритм для оптимізації шляху. Досліджено ефективність федеративного навчання FedProx для колективного вдосконалення моделей у рої БПЛА. Результати підтверджено шляхом моделювання у середовищі Gazebo/ROS 2.

КЛЮЧОВІ СЛОВА: БПЛА, РІЙ, FMCW РАДАР, УНИКНЕННЯ ЗІТКНЕНЬ, ІНТЕЛЕКТУАЛЬНІ МЕТОДИ, ПРОГНОЗУВАННЯ ТРАЄКТОРІЙ, ATTENTION-BI-GRU, АЛГОРИТМ МУРАШИНОЇ КОЛОНІЇ, FEDPROX, GAZEBO, РОБОТОТЕХНІКА.

ABSTRACT

The part of the master's qualification work: 109 pages, 37 pictures, 4 tables, 44 sources.

The purpose of the work – to increase the reliability of autonomous UAV navigation by implementing an intelligent decision-making model for collision avoidance based on radar data.

Object of research – safety of autonomous swarm UAV flights in the presence of dynamic obstacles.

Subject of research – methods and algorithms for collision avoidance using on-board radar systems and artificial intelligence.

Summary of the work: In the qualifying master's thesis, an analysis of decentralized navigation architectures and FMCW radar signal processing technologies was carried out. A hybrid methodology has been developed that combines the Attention-Bi-GRU neural network for trajectory prediction and the ant colony algorithm for path optimization. The effectiveness of FedProx federated learning for collective model improvement in a UAV swarm is investigated. The results are confirmed by simulation in the Gazebo/ROS 2 environment.

KEYWORDS: UAV, SWARM, FMCW RADAR, COLLISION AVOIDANCE, INTELLIGENT METHODS, TRAJECTORY PREDICTION, ATTENTION-BI-GRU, ANT COLONY OPTIMIZATION, FEDPROX, GAZEBO, ROBOTICS.

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут Телекомунікацій

ПОДАННЯ
ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступня магістра

Направляється здобувач Бовсунівський М.С. до захисту кваліфікаційної роботи
(прізвище та ініціали)

за спеціальністю 172 Електронні комунікації та радіотехніка
(шифр і назва спеціальності)

Освітньо- професійної програми Телекомунікаційні системи та мережі

на тему: « Інтелектуальні методи уникнення зіткнень у групових польотах БПЛА на основі РЛС».

Кваліфікаційна робота і рецензія додаються.

Директор ННІТ _____ Владислав Кравченко
(підпис) (Ім'я, ПРІЗВИЩЕ)

Висновок керівника кваліфікаційної роботи

Здобувач Бовсунівський Микола Сергійович показав гарну теоретичну та інженерну підготовку, уміння володіти новими комп'ютерними технологіями, користуватися навчальною, довідковою і науково-технічною літературою, в тому числі рекомендаціями МСЕ. Працюючи над завданнями, які доручались керівником, проявив ініціативність, сумлінність та хист до інженерної роботи.

Все це дозволяє оцінити виконану кваліфікаційну роботу здобувача Бовсунівського Миколи Сергійовича на оцінку "відмінно" та присвоїти йому кваліфікацію «магістр з електронних комунікацій та радіотехніки».

Керівник роботи _____ Віктор ЗАЇКА
(підпис) (Ім'я, ПРІЗВИЩЕ)

" " _____ 2025 року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач Бовсунівський М.С. допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедри Телекомунікаційних систем та мереж _____ Віктор ЗАЇКА
(назва) *(підпис)* *(Ім'я, ПРІЗВИЩЕ)*

" " _____ 2025 року

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну роботу
на здобуття освітнього ступеня магістра

здобувача вищої освіти Бовсунівського Миколи Сергійовича

на тему: “Інтелектуальні методи уникнення зіткнень у групових польотах БПЛА на основі РЛС”

Актуальність:

Масове впровадження роїв БПЛА для моніторингу та пошуково-рятувальних операцій потребує високого рівня безпеки польотів. Ключовою проблемою є запобігання зіткненням в умовах обмеженої видимості (дим, туман), де оптичні датчики неефективні. Використання FMCW-радарів у поєднанні зі штучним інтелектом для створення автономних систем групового керування є актуальним науковим завданням.

Позитивні сторони:

Робота дозволяє досить повно оцінити загальну характеристику, сутність та структуру поставленої науково-технічної проблеми. У вступній частині дається чітке визначення мети, об’єкту та предмету дослідження, коректно сформульовано завдання та обрано методику їх вирішення.

У роботі проведено глибокий аналіз децентралізованих навігаційних архітектур та розроблено інноваційну гібридну методологію керування, що базується на інтеграції обробки сигналів FMCW-радарів із сучасними нейромережами архітектури Attention-Bi-GRU. Особливу увагу приділено науковій новизні, яка обґрунтована впровадженням механізму федеративного навчання FedProx для автономного самонавчання систем штучного інтелекту безпосередньо на бортових обчислювачах БПЛА.

Практична частина роботи містить об’єктивне підтвердження функціональності запропонованих алгоритмів шляхом їх тестування у високотехнологічному середовищі моделювання Gazebo/ROS 2. Отримані результати щодо зниження ймовірності зіткнень підтверджують високу практичну цінність розробки для сучасних інфокомунікаційних систем.

Зміст магістерської кваліфікаційної роботи повністю відповідає виданому завданню, а дослідження виконано у повному обсязі з дотриманням вимог наукового стилю.

Недоліки:

1. Недостатньо висвітлено питання стійкості системи до хибних сигналів радара або похибок прогнозування нейронної мережі.
2. Зустрічаються деякі друкарські та стилістичні помилки.

Висновок: кваліфікаційна робота на здобуття ступеня магістра заслуговує оцінку «відмінно», а здобувач Бовсунівський Микола Сергійович заслуговує присвоєння кваліфікації : магістр з електронних комунікацій та радіотехніки.

Рецензент: _____ _____

Науковий ступень, вчене звання підпис Ім’я, ПРІЗВИЩЕ

ЗМІСТ

ВСТУП 11

РОЗДІЛ 1 ОГЛЯД СУЧАСНИХ МЕТОДІВ УНИКНЕННЯ ЗІТКНЕНЬ ДЛЯ ГРУП БПЛА 14

1.1. Класифікація методів уникнення зіткнень 14

1.1.1. Основні виклики та обмеження в умовах щільних роїв 17

1.2. Використання FMCW радарів у навігації БПЛА 17

1.3.1. Критерії оцінки якості алгоритмів планування шляху 25

1.3.2. Обмеження та перспективи інтелектуальних алгоритмів планування у динамічних роях 25

1.4. Нейромережі для прогнозування траєкторій 27

1.4.1. Формалізація завдання прогнозування траєкторій 31

1.5. Архітектури систем навігації БПЛА 31

1.5.1. Вимоги до комунікаційних протоколів 34

Висновки до першого розділу та постановка проблеми 35

РОЗДІЛ 2 ТЕОРЕТИЧНЕ ОБҐРУНТУВАННЯ ТА МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ СИСТЕМИ УНИКНЕННЯ ЗІТКНЕНЬ 37

2.1. Математична модель руху БПЛА 38

2.4.1. Взаємодія модулів системи 48

2.4.2. Вимоги до децентралізованої роботи 50

2.5. Модуль обробки даних FMCW-радара та оцінювання стану 51

2.5.1. Алгоритм фільтрації та кластеризації 51

2.5.2. Застосування розширеного Калманівського фільтра (EKF) 52

2.6. Модуль прогнозування траєкторій на основі нейромережі Attention-Bi-GRU 55

2.6.2. Підготовка даних та навчання 57

2.6.3. Вихідні дані модуля 57

2.7. Модифікований алгоритм мурашиних колоній (ACO) для планування безпечного шляху 58

2.7.1. Адаптація до безперервного простору 59
2.7.2. Інтеграція прогнозів в евристику 60
2.7.3. Врахування обмежень 61
2.7.4. Децентралізована оптимізація 63
2.8. Механізм адаптації та покращення через федеративне навчання (FedProx) 63
2.8.1. Принципи федеративного навчання 63
2.8.2. Реалізація FedProx 64
2.8.3. Стратегія обміну 65
2.9 Алгоритм роботи повного циклу системи AI-RDR 66
2.9.1. Псевдокод основного алгоритму 66
РОЗДІЛ 3 МОДЕЛЮВАННЯ ТА АНАЛІЗ ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОЇ СИСТЕМИ AI-RDR 72
3.1. Середовище та інструменти моделювання 72
3.2. Параметри моделювання та сценарії експериментів 73
3.3. Метрики ефективності 75
3.4. Результати моделювання та порівняльний аналіз 76
3.5. Обговорення результатів та обмеження 79
РОЗДІЛ 4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ 81
4.1. Аналіз витрат на апаратно-програмну реалізацію системи AI-RDR 81
4.1.1. Вибір та вартісна оцінка апаратних компонентів 81
5.1.2. Аналіз експлуатаційних витрат та впливу на льотні характеристики 82
Вплив на масу та льотні характеристики. 82
4.2. Оцінка економічного ефекту та ефективності впровадження 84
4.2.1. Оцінка потенційних збитків від зіткнення 84
4.3. Аналіз конкурентоспроможності та ринкових перспектив 86
Висновки до четвертого розділу 88
ВИСНОВКИ 89

ПЕРЕЛІК ПОСИЛАНЬ 91

ДОДАТКИ 98

Додаток А 98

Додаток Б 99

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ 100

ВСТУП

Актуальність теми. Швидка еволюція БПЛА, і особливо технологій ройових польотів, відкриває нові можливості в кількох галузях –від моніторингу навколишнього середовища та пошуку та порятунку до логістики та оборони. Однак, перешкодою для масового впровадження групових польотів все ще залишається гарантоване уникнення зіткнень у щільних динамічних середовищах.

У рої кожен пристрій діє як рухома перешкода для інших; таким чином, він утворює вже складну динамічну систему з високими ризиками/випадками зіткнення.

Штучні потенційні поля або алгоритми перешкод швидкості –традиційні методи швидкого виправлення для уникнення зіткнень –не можуть забезпечити безпеку в роях. Їхня реактивність, локальні мінімуми, до яких вони так легко спускаються, та припущення лінійності руху сусіда призводять до небезпечної конвергенції –точок зупинки з різким падінням ефективності. Тому потрібен підхід, який може бути проактивним (запобігати, перш ніж це станеться), адаптивним до динамічної поведінки групи.

Сучасний розвиток сенсорних технологій та штучного інтелекту створює передумови для таких рішень. Зокрема, компактні повітряні радари FMCW (радари з лінійною частотною модуляцією) забезпечують стабільне вимірювання дальності, швидкості та азимуту цілей незалежно від освітлення чи погодних умов. Інтелектуальна обробка цих даних, наприклад, за допомогою рекурентних нейронних мереж з механізмом уваги (Attention-Bi-GRU), дозволить виявити прогнозування траєкторій сусідніх БПЛА. Такі метаевристичні алгоритми планування, як алгоритм колонії мурах (ACO), пов'язують прогнозовану інформацію для створення гібридної системи, здатної не тільки реактивно, але й проактивно запобігати зіткненням.

*Мета і задачі дослідження.*Метою цієї магістерської кваліфікаційної роботи є розробка гібридного інтелектуального методу для групових польотів БПЛА на основі даних бортового радара FMCW, який поєднує точне зондування, прогнозування нейронної мережі та колективну оптимізацію траєкторії.

Основними цілями, поставленими в рамках цієї роботи, є: проведення огляду та аналізу існуючих технічних рішень у галузі бортових радарів БПЛА, а також методів, що використовуються для їх даних; розробка вимог до радіолокатора-далекоміра, які будуть синтезовані далі в середовищі програмного забезпечення в циклі (SITL) з урахуванням відомих обмежень, що накладаються реалізаціями фізичного двигуна.

Для досягнення цієї мети в роботі були поставлені та вирішені наступні завдання: проаналізувати сучасні методи запобігання зіткненням та обґрунтувати вибір гібридного методу; дослідити можливості використання моделей радіолокаційного моделювання FMCW та нейронних мереж (Attention-Bi-GRU) для прогнозування руху рою; розробити архітектуру системи, що інтегрує радіолокаційне сприйняття, модуль прогнозування та модифікований алгоритм ACO для планування безпечного шляху; реалізувати програмний прототип системи та дослідити можливість її вдосконалення за допомогою федеративних механізмів навчання (FedProx); провести експериментальну перевірку ефективності запропонованої системи в середовищі моделювання та порівняння з базовими підходами.

Об'єкт дослідження –процес забезпечення безпеки групи шляхом уникнення зіткнень.

Предмет дослідження –методи та алгоритми інтелектуального запобігання зіткненням на основі радіолокаційних даних, зокрема гібридні методи, що інтегрують прогнозування нейронних мереж (Atenntion-Bi-GRU) та оптимізаційне планування (ACO).

Наукова новизна, як доведено отриманими результатами, полягає в розробці синергетичної архітектури між проактивною системою запобігання зіткненням, яка вперше об'єднує надійне радіолокаційне сприйняття (FMCW), інтелектуальне прогнозування траєкторії за допомогою Attention-Bi-GRU та децентралізоване планування на основі модифікованого алгоритму мурашиної колонії (ACO) в один цикл прийняття рішень. Запропоновано інтеграцію, в якій прогнози з нейронних мереж безпосередньо вбудовуються в евристичну функцію ACO, тим самим виконуючи евристичне планування на основі пошуку з урахуванням прогнозованих майбутніх станів навколишнього середовища. Для прикладу задачі запобігання

зіткненням рою, що розглядається в цій роботі, вперше було застосовано/знайдено/перевірено алгоритм федеративного навчання FedProx для колективної адаптації прогнозних моделей в умовах децентралізованих та гетерогенних даних.

Практичне значення одержаних результатів. Включають функціонуючий програмний прототип системи, здатний інтегруватися з реальними платформами БПЛА. Результати імітаційного моделювання доводять, що запропонований підхід підвищує безпеку групових польотів шляхом збільшення мінімальних дистанцій ешелонування та зменшення кількості небезпечних підходів, які потрібно враховувати, порівняно з традиційними методами. Розроблена децентралізована архітектура є більш масштабованою, не потребує централізованого управління для її впровадження, що тим самим спрощує її впровадження.

Апробація результатів дослідження

Основні положення та результати магістерської роботи доповідалися, обговорювалися та отримали позитивну оцінку на:

- Науково-технічних конференціях студентів, аспірантів та молодих учених Державного університету інтелектуальних технологій і зв'язку (м. Київ, 2024–2025 рр.);
- Науково-практичних семінарах кафедри телекомунікаційних систем та технологій, присвячених питанням автоматизації управління безпілотними апаратами та впровадженню інтелектуальних методів обробки радіолокаційних сигналів.

Структура та обсяг роботи

Робота складається зі вступу, 5 розділів, загальних висновків, списку використаних джерел та додатків. Повний обсяг пояснювальної записки становить 109 сторінок комп'ютерного тексту. Робота містить 37 рисунків, 4 таблиці та 2 додатки. Список використаних джерел включає 44 найменування.

РОЗДІЛ 1
ОГЛЯД СУЧАСНИХ МЕТОДІВ УНИКНЕННЯ ЗІТКНЕНЬ ДЛЯ ГРУП БПЛА

1.1. Класифікація методів уникнення зіткнень

Методи можна класифікувати як реактивні, проактивні або комбінацію обох гібридних [1, 2]. Визначення, дане терміну «реактивний» у цьому контексті, описує методи, які приймають рішення на основі поточного стану середовища та агентів без будь-якого прогнозування майбутніх дій сусідів. Були наведені приклади алгоритмів штучного потенційного поля APF та геометричних методів, заснованих на перешкодах швидкості VO [3]. В APF БПЛА змушується рухатися під дією віртуальних сил, де ціль притягується, а перешкоди відштовхуються; траєкторія розраховується як градієнтний загальний потенціал. Однак цей простий та швидкий підхід має суттєві недоліки: можуть з'явитися локальні мінімуми потенціалу, що зупиняє БПЛА до досягнення цілі, більше того, якщо сила відштовхування домінує над дуже близькою перешкодою, він може стати недосяжним. Крім того, той факт, що APF є чисто реактивним без урахування динаміки, призводить до коливань та неоптимальних «рваних» траєкторій поблизу перешкод. Методи VO (включаючи вдосконалені версії RVO, ORCA) формалізують умови для уникнення зіткнень у просторі швидкостей [3, 4].

Наприклад, ORCA визначає дозволену область швидкостей для кожного агента таким чином, що мінімальне відхилення від бажаної траєкторії забезпечується шляхом генерації плавних шляхів. Однак, основне припущення, зроблене методами VO –сусіди продовжуватимуть рух зі своєю поточною швидкістю –майже ніколи не є істинним у реальних роях. Тому реактивні алгоритми на основі VO є надмірно консервативними (уникають зіткнень, яких би не сталося) і навіть досягають глухого кута при високій щільності рою.

Проактивні методи спрямовані на подолання обмежень реактивних методів шляхом прогнозування розвитку ситуації в русі попереду.[5] На відміну від алгоритму, який реагує лише на положення та швидкості, на даний момент моделі руху використовуються проактивно в системах, де майбутні шляхи сусідів можна систематично передбачити, щоб відповідно спланувати свій шлях; таким чином, встановлення рішення з припущенням щодо подальшого розвитку стану підвищує надійність запобігання зіткненням у динамічному середовищі. Прикладами проактивних підходів є методи з плануванням траєкторії на заданому часовому горизонті, використання моделювання поведінки сусідів (наприклад, лінійних або нелінійних моделей руху), а також алгоритмів на основі намірів прогнозування [6]. Зокрема, вдосконалений підхід V-RVO запропонував модифікацію VO шляхом включення прискорень агентів [7], а робота [8] поєднує перешкоди швидкості з функціями контролю бар'єрів, щоб гарантувати безпечний рух кількох БПЛА. Проактивні методи зазвичай призводять до більш стабільної поведінки рою, але вимагають точного прогнозування і тому часто є ще більш обчислювально складними.

Гібридні методи інтегрують переваги як реактивних, так і проактивних підходів. Такі системи поєднують швидкість реактивної реакції на непередбачувані перешкоди з проактивним плануванням шляху з урахуванням прогнозованих змін [9]. Гібридний підхід може, наприклад, використовувати надійні дані датчиків для негайного уникнення зіткнення, одночасно плануючи оптимальний маршрут з урахуванням прогнозованих траєкторій сусідніх БПЛА. Сучасні дослідження підтверджують ефективність переходу до проактивно-реактивних алгоритмів: успішна система уникнення зіткнень повинна не тільки реагувати на поточне розташування перешкод, але й виконувати ймовірнісне уникнення траєкторій, які можуть виникнути найближчим часом [5, 6]. Іншими словами, підвищення точності прогнозування руху агента безпосередньо сприяє побудові безпечніших та ефективніших маршрутів у рої. У таблиці 1.1 підсумовано ключові характеристики реактивних, проактивних та гібридних методів уникнення зіткнень.

Таблиця 1.1

Класифікація методів уникнення зіткнень

Категорія	Приклади підходів	Переваги	Недоліки
Реактивні	APF, VO/ORCA, правил-базовані	Швидка реакція в реальному часі; простота реалізації	Локальні екстремуми (APF); ігнорують майбутню динаміку сусідів; можуть бути надто консервативними
Проактивні	Планування траєкторії з прогнозом, моделі намірів	Урахування майбутніх змін дає стабільніші, гладкі траєкторії; вищий рівень безпеки у динамічному середовищі	Більша обчислювальна складність; залежність від точності моделей прогнозу
Гібридні	Комбіновані (сенсори + AI-прогноз + локальне уникнення)	Поєднання швидкості реакції та завчасного планування; робастність до різних ситуацій	Складність реалізації; потреба у координації між компонентами (сенсори, алгоритми прогнозу і планування)

Як видно з таблиці, реактивні алгоритми добре працюють для простих сценаріїв, а також демонструють низьку затримку, але стають абсолютно неефективними в деяких складних ройових формуваннях. Проактивні забезпечують кращу адаптацію до динаміки, але тут знову виникає вимога до потужного обчислювального ресурсу з точною моделлю руху. Гібридні –це нова тенденція, яка намагається успадкувати найкращі риси обох підходів [9].

Враховуючи вимоги до роїв БПЛА –децентралізованих, масштабованих та безпечних у невідомому середовищі –інтелектуальні гібридні алгоритми все ще є найперспективнішими для повністю автоматичної реалізації запобігання зіткненням.

1.1.1. Основні виклики та обмеження в умовах щільних роїв

Специфічні проблеми, пов'язані з проблемами тупикових блокувань, виникають при розробці методів, особливо застосовних до щільних роїв, що тим самим обмежує застосовність класичних підходів:

Проблема тупикових блокувань: У ситуаціях високої щільності реактивні методи, особливо ті, що базуються на жорстких правилах або APF, можуть призвести до взаємного блокування агентів, оскільки кожен намагається уникнути іншого, але в результаті система переходить у стаціонарний стан, з якого неможливо вийти без централізованого втручання [2, 4].

Нестабільність та коливання: коливання траєкторії польоту кількох БПЛА виникають як наслідок постійної зміни курсу відносно інших агентів поблизу. Це призводить до втрат енергії через небажані коливання разом із загальною непередбачуваною поведінкою [3, 9].

Обчислення в масштабі: Складність проактивних алгоритмів (алгоритм, який передбачає K кроків наперед для N сусідів) є нелінійною в $O(N^2)$ або гірше. Тому такі алгоритми ніколи не підійдуть для великих роїв, що працюють на бортовому комп'ютері з обмеженою потужністю [5,10].

Вимоги до зв'язку: У деяких децентралізованих підходах необхідна повна або часткова взаємодія між агентами. Це стає серйозною проблемою в практичних середовищах, оскільки затримка зв'язку в реальному часі, обмеження пропускної здатності та втрата пакетів [11, 12] ще не враховані.

Вплив неточностей датчиків: Будь-який метод залежить від точності, з якою можна виміряти положення та швидкість сусідів. Помилки найбільше накопичуються в прогнозованих траєкторіях і призводять або до необґрунтованого ризику, або до надмірного консерватизму [6, 13].

1.2. Використання FMCW радарів у навігації БПЛА

Для надійного виявлення перешкод та інших дронів у рої вибір бортових датчиків є критично важливим. Останнім часом велику увагу привернули частотно-модульовані радари безперервної хвилі (FMCW), які забезпечують сприйняття

навколишнього середовища за будь-яких погодних умов та за будь-якого сезону [10]. На відміну від оптичних камер або лідарів, умови освітлення не впливають на радарні датчики; вони чудово працюють у тумані, дощі, пилу або умовах низької видимості. Це робить радари незамінними для безпечної роботи роїв саме в тих сценаріях, коли інші датчики стають ненадійними. Сучасні компактні радарні модулі дозволяють оснащувати радарами навіть відносно невеликі дрони, що збільшує функціональність, доступну рою (політ вночі, в диму або в ситуаціях, коли GPS відсутній) [10]. Зокрема, радар FMCW вважається оптимальним вибором для навігаційних завдань та завдань запобігання зіткненням як на ближній, так і на середній відстані в межах щільного рою.

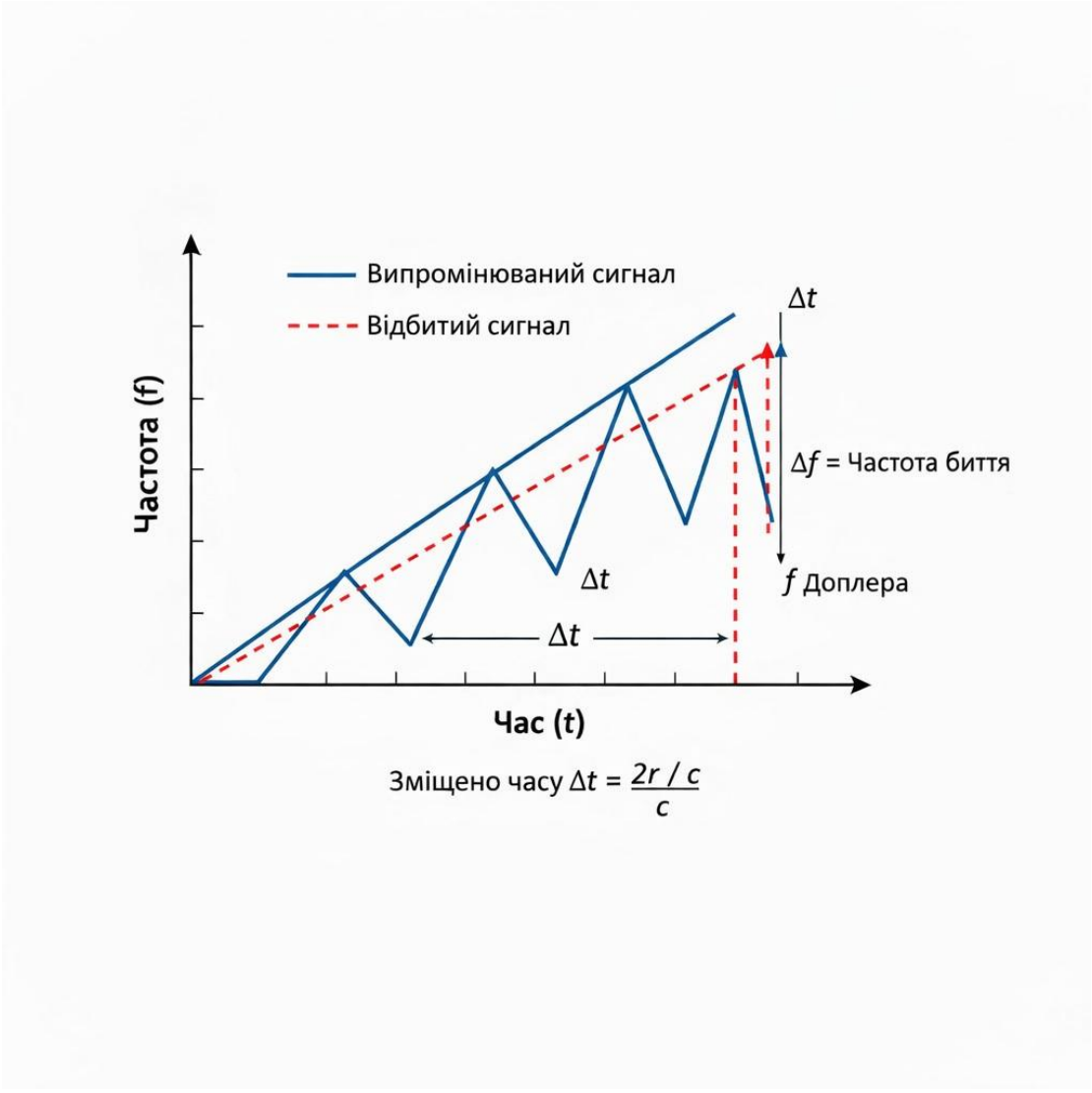


Рис. 1.1 Принцип дії FMCW радара.

Графік показує лінійно змінний сигнал, що випромінюється радаром (суцільна лінія), та сигнал, відбитий від цілі (пунктирна лінія), зміщений у часі на Δt. Цей зсув відповідає подвоєній дальності до цілі (2r), поділеній на швидкість поширення радіохвиль (c). За різницею частот між переданим та прийнятим сигналами (так звана частота биття) радіолокатор FMCW визначає відстань до об'єкта, а за величиною доплерівського зсуву частоти - відносну швидкість цілі [10, 11].

Основний принцип FMCW-радара полягає в передачі безперервної хвилі, частота якої лінійно змінюється з часом, та одночасному прийомі відбитих сигналів. Різниця частот між переданим та прийнятим сигналами дає час затримки Δt відбитого сигналу, за яким можна визначити відстань до відбивача. Крім того, порівняння частоти приймального сигналу на висхідній та низхідній ділянках модуляції дозволяє оцінити доплерівський зсув, пропорційний швидкості цілі відносно БПЛА, таким чином FMCW-радар вимірює як дальність, так і відносну швидкість цілей [10][11]. Він виводить тривимірні дані для кожної виявленої цілі завдяки багатопроменевому приймачу: дальність; радіальна швидкість; азимут (кутове положення).

Таблиця 1.2

Порівняльні характеристики датчиків для навігації БПЛА

Датчик	Принцип дії	Переваги	Недоліки	Придатність для щільних роїв
1	2	3	4	5
Оптична камера (RGB/Stereo)	Зйомка зображень, стереозір	Висока роздільна здатність, текстурна інформація, дешевизна	Залежність від освітлення, погоди; великий обсяг даних; складність обробки	Обмежена (погода, умови)
Лідар (LiDAR)	Лазерне сканування	Точні 3D-дані (хмара точок), незалежність від освітлення	Висока вартість, великі розміри/вага, чутливість до пилу/туману	Середня (вартість, розмір)

Продовження таблиці 1.2

1	2	3	4	5
Ультразвук	Відлуння звукової хвилі	Дешевий, точний на малих відстанях, простий	Мала дальність, вразливість до акустичних перешкод, низька швидкість	Низька (дальність)
FMCW Радар	Частотна модуляція радіохвилі	Незалежність від погоди, пряме вимірювання швидкості, компактність, середня дальність	Нижча кутова роздільність, можливі електромагнітні перешкоди	Висока (всепогодність, швидкість)

Основними перевагами FMCW-радарів є висока точність на коротких та середніх відстанях, невеликі розміри, а також відносно низька вага та енергоспоживання (порівняно з імпульсними радарами) [10]. Висока роздільна здатність та чутливість завдяки сучасній цифровій обробці сигналів дозволяють розрізняти навіть дуже малі об'єкти. FMCW-радари цілком підходять для застосування в уникненні перешкод у реальному часі, наприклад, коли дрон оглядає лінії електропередач, вітрові турбіни або будь-які інші елементи інфраструктури, він може наблизитися на безпечну відстань на основі показань радара [10]. Ще однією перевагою є те, що вони працюють у мікрохвильовому діапазоні частот незалежно від GPS або будь-яких зовнішніх систем, таким чином, радари можуть самостійно орієнтуватися в просторі.

Основним недоліком радарів FMCW є їхня відносно мала дальність дії порівняно з радарами з великою імпульсною або синтезованою апертурою (SAR)[10]. Однак це не стає серйозною проблемою в умовах ройового скупчення, оскільки відстані між кожним дроном мінімальні –від десятків до сотень метрів. Система зазнаватиме перешкод, якщо багато радарів працюють одночасно

поблизу; тому для вирішення цієї конкретної проблеми можна використовувати різні діапазони частот або сигнали з часовим поділом. Слід також враховувати, скільки потоку даних генерує бортовий радар, який потребує швидкої обробки бортовим комп'ютером, але в наш час цифрові процесори сигналів мають достатні можливості для обробки таких ситуацій. З усіма цими незначними відмінностями сьогодні радар FMCW вважається основним датчиком, що забезпечує надійне запобігання зіткненням між БПЛА, що летять групою.[10, 12] Цей датчик виявляє постійний потік інформації про об'єкти навколо, незалежно від будь-яких зовнішніх умов, тому на ньому базуються інтелектуальні алгоритми прогнозування та планування маршруту.

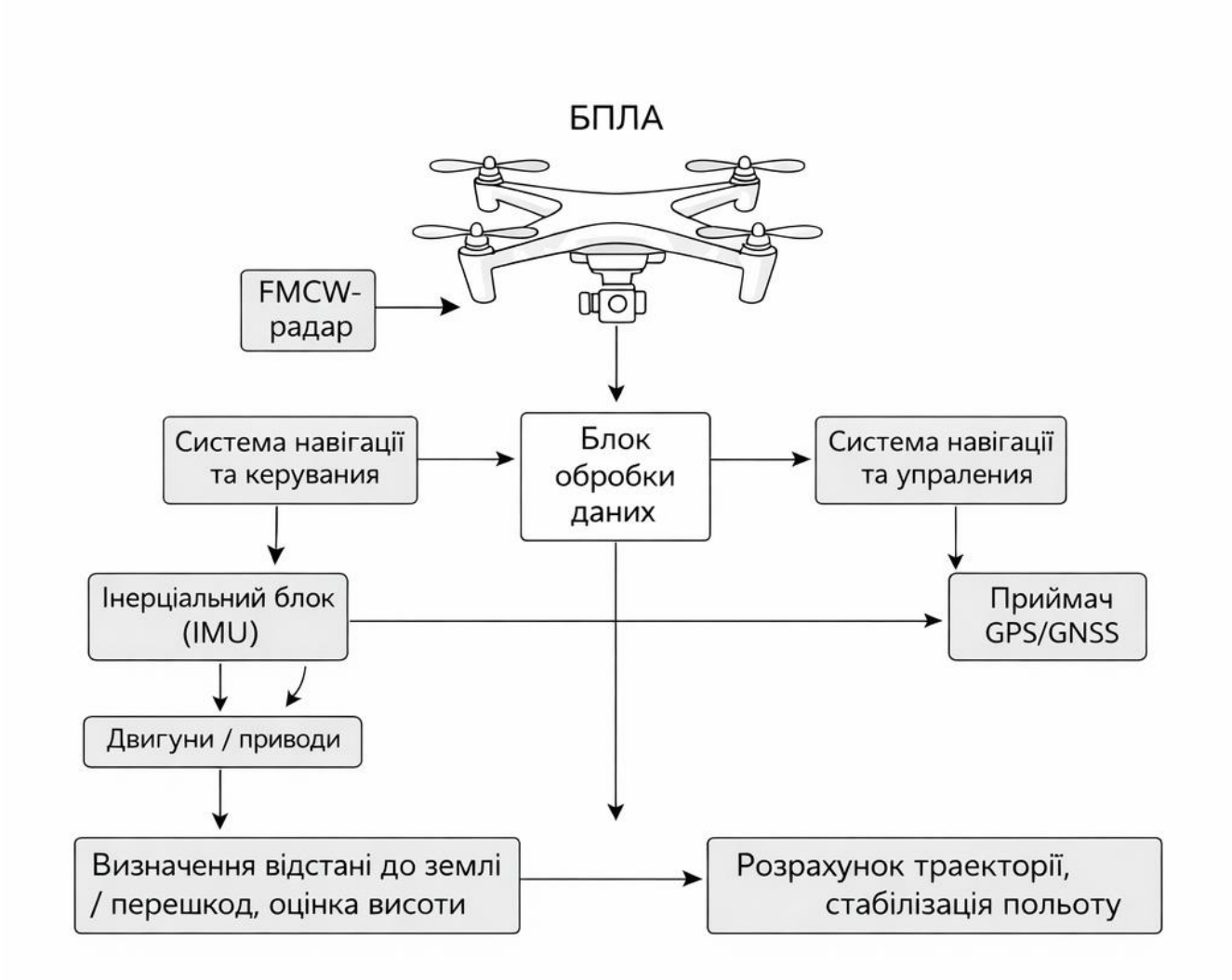


Рис. 1.2 Схема інтеграції FMCW радара в систему навігації БПЛА.

1.3. Інтелектуальні алгоритми планування шляху для роїв

Глобально оптимальна траєкторія маршруту для рою, яка включатиме глобальну оптимізацію маршрутів для набору рухомих агентів, разом з алгоритмами, що підтримують розподілене прийняття рішень, може бути досягнута лише за допомогою такого підходу. Класичні детерміновані підходи (наприклад, однакові потенційні поля або VO з розділу 1.1) навіть не забезпечують оптимальності, а також обтяжені раніше згаданими проблемами масштабованості.[13] Тому в цій роботі пропонуються метаевристичні алгоритми, що або імітують колективну поведінку, що зустрічається в природі, або застосовують еволюційні принципи, оскільки оптимальні рішення не можуть бути гарантовані в розумні терміни жодним існуючим алгоритмом через складність проблеми. Найпопулярнішими серед них є оптимізація колонії мурах (ACO), оптимізація рою частинок (PSO), генетичний алгоритм (GA) штучні потенційні поля (APF), що покращуються динамічно/адаптивно.

Оптимізація колонії мурах (ACO) –це метаевристичний метод, натхненний поведінкою мурах у пошуку шляхів від їхньої колонії до джерел їжі та назад, як пояснюється в [14]. У рамках планування маршруту БПЛА ACO застосовує віртуальні феромонні сліди, залишені успішними мурахами-агентами протягом кількох ітерацій побудованих траєкторій. Його переваги включають паралелізм – багато агентів можуть досліджувати простір рішень одночасно –та майже глобальну оптимальність.[15] Пошук шляху для динамічних цілей дуже добре підходить для області застосування, яка довела свою ефективність для цього алгоритму: особливо при використанні з БПЛА в умовах рухомих перешкод[14]. Класична дискретна версія була модифікована для застосування її в безперервному 3D-просторі; замість вибору наступної дискретної вершини шляху, «мураха» генерує напрямок наступної точки з безперервного розподілу, сформованого на основі карти феромонів найкращих рішень. Ключовим аспектом сучасних алгоритмів є інтеграція евристик безпеки, які враховують прогнозовані ризики зіткнень, оцінюючи кожен крок на траєкторії.[5] наведено приклад, у якому до

евристичної функції для ACO додається оцінка ймовірності конфлікту: шляхи генеруються таким чином, щоб уникнути областей, де, згідно з прогнозуванням траєкторій сусідів, існує висока ймовірність появи інших агентів. Цей підхід представляє синергію як планування, так і прогнозування, що дозволяє ініціаторам проактивно вирішувати потенційні зіткнення до таких контактів.

Оптимізація рою частинок (PSO) належить до категорії класу біологічно натхненних методів оптимізації, які імітують колективну поведінку птахів, що збираються в зграї, або риб, що збираються у великій кількості [15]. По суті, PSO реалізує правило оновлення для кандидатів на рішення в бік кращих кандидатів відповідно до певних критеріїв якості та одночасно застосовує це оновлення до кількох кандидатів (частинок). Зміна стану включає власне найкраще знайдене рішення частинки та найкраще в глобальному масштабі серед усіх знайдених на сьогодні рішень. Простота як у реалізації, так і в конфігурації робить PSO привабливим для застосувань планування траєкторії БПЛА - алгоритм не вимагає жодних складних моделей навколишнього середовища; достатньо мати функцію, яка оцінює якість траєкторії (наприклад, довжину шляху зі штрафами за відстані поблизу перешкод) [15]. Однак, це також має негативну сторону, яка полягає в тому, що відбувається передчасна конвергенція - швидке накопичення частинок у деякому локальному оптимумі без перевірки інших областей простору [15]. Щоб запобігти цьому, адаптивна стратегія динамічно змінює коефіцієнт інерції, щоб підтримувати належний баланс між дослідженням та експлуатацією [15].

Генетичні алгоритми (ГА) застосовуються до планування траєкторії шляхом еволюційного пошуку: набір можливих шляхів кодується як «хромосоми», і вони проходять операції відбору, кросинговеру та мутації [13]. Успішна реалізація призводить до ефективного уникнення локальних мінімумів, оскільки вона поєднує вдалі сегменти з різних рішень. В області роїв БПЛА генетичний алгоритм використовувався, зокрема, для оптимізації маршрутів доставки вантажів дронами [13], а також застосовувався для багатоцільового покриття області. Гнучкість належить до списку переваг цього методу - функцію придатності форми легко

можна змінювати з урахуванням різних критеріїв (довжина шляху, споживання енергії, безпека зіткнень тощо). Відомо, що близькість до глобального оптимуму в складних просторах ГА знаходить рішення, але збільшення проблеми розмірності призводить до складних обчислень. Крім того, щоб еволюція розвивалася успішно і не прийшла до стану стагнації, параметри алгоритму повинні бути встановлені правильно (ймовірності кросинговеру, мутації, розмір популяції).

Штучні потенційні поля (ШПП) –це класичний реактивний метод, який додає більше покращень для використання в динамічних середовищах. Сучасні ШПП можуть мати змінні вагові коефіцієнти, які коригують значення залежно від ситуації навколишнього середовища або поєднуються з іншими алгоритмами (наприклад, поєднання ШПП як локального коректора траєкторії з глобальним планувальником на основі ACO або GA)[1]. Такі гібридні схеми розроблені для нейтралізації недоліків ШПП –наприклад, додавання випадкових збурень або детектора перешкод допомагає боротися з локальними мінімумами. Дослідження також показують успіх спільного використання нечіткої логіки та ШПП для динамічного коригування параметрів поля відповідно до ситуації[16]. Тож, по суті, швидкі засоби уникнення перешкод, такі як інтелектуальна надбудова над роями, все ще можуть використовувати цей інструмент.

Фундаментально, розподілене прийняття рішень –єдиний спосіб зробити великі рої БПЛА життєздатними.[8,17] Централізоване планування маршрутів в одній обчислювальній точці для всіх дронів має вразливості, оскільки, по-перше, може бути забезпечено обмежене масштабоване рішення, а по-друге, відмова центрального вузла паралізує весь рій. Як визначили Санчес та ін.[8], повністю централізований контролер стає «вузьким місцем» у комунікації та схильний до збоїв. Тому за цією архітектурою кожен БПЛА локально планує свій маршрут на основі інформації від своїх датчиків та, можливо, найближчих сусідів. Розподілені реалізації метаевристик, що обмінюються інформацією між сусідами, такі як ACO або PSO, природно вписуються в таку архітектуру: у цьому випадку кожен дрон діє як агент оптимізації, який обмінюється з іншими агентами мінімально необхідними

даними (наприклад, оцінений оптимальний шлях або «феромонний» показник безпеки маршруту) [5,14,15]. Саме тому він залишається високомасштабованим та стійким до збоїв або відмов окремих агентів, оскільки рій в цілому продовжує функціонувати, навіть якщо деякі агенти виходять з ладу, а також тому, що обчислювальне навантаження розподіляється між усіма БПЛА. Останні тенденції в дослідженнях планування шляхів для роїв сприяють такому способу роботи, за якого рішення знаходять спільно та децентралізовано в межах самого рою, завдяки чому адаптація рою до змінних умов відбувається автоматично, без нав'язування своєї волі будь-якою окремою точкою [8, 17].

1.3.1. Критерії оцінки якості алгоритмів планування шляху

Для об'єктивного порівняння алгоритмів планування траєкторії в контексті щільних роїв використовуються такі ключові метрики:

Довжина шляху, час досягнення цілі, споживання енергії (інтеграл від споживаної потужності). Мінімальна відстань до найближчої перешкоди/БПЛА протягом усього польоту, ймовірність зіткнення, кількість «небезпечних підходів». Максимальне прискорення та ривок, що впливає на стійкість та комфорт польоту (для вантажу).

Обчислювальна ефективність: Середній час обчислення однієї траєкторії, максимальне навантаження на процесор, споживання пам'яті.

Масштабованість: Зміна ефективності та обчислювальної складності зі збільшенням кількості агентів у рої (N) та щільності середовища.

1.3.2. Обмеження та перспективи інтелектуальних алгоритмів планування у динамічних роях

Навіть з інтелектуальними та метаевристичними алгоритмами планування шляху, практичні обмеження, що застосовуються до середовища, де рій БПЛА є одночасно щільним та динамічним, переважають їхню теоретичну перевагу в ефективності. Основна проблема полягає в залежності хороших планів від повної та точної інформації про майбутній стан середовища. Більшість класичних

реалізацій ACO, PSO та генетичних алгоритмів базуються на певних припущеннях щодо квазістатичних перешкод або просто використовують інформацію про поточний стан від сусідів; отже, вони стають вкрай неефективними в ситуаціях, що включають раптові маневри або високу щільність агентів у просторовій області.

Ще одне дуже важливе обмеження стосується аспекту часу на планування. Метаевристичні алгоритми, хоча й здатні забезпечувати приблизно оптимальні рішення, вимагають ітеративного процесу пошуку, який створює компроміс між якістю траєкторії та допустимим часом обчислення. У застосунках реального часу, особливо на борту БПЛА з обмеженими обчислювальними ресурсами, це означає зменшення кількості ітерацій або використання спрощених моделей, що знову ж таки призводить нас до неоптимальних або надмірно консервативних рішень.

Також слід особливо зазначити невизначеність руху сусідів. Під час польоту рою кожен агент не лише реагує на інших, але й проактивно змінює середовище. Відсутність будь-якого механізму прогнозування поведінки сусідніх БПЛА призводить до ситуацій, коли навіть глобально оптимальний план траєкторії швидко застаріває. Це особливо стосується алгоритмів, заснованих на деякій історично накопиченій інформації, наприклад, картах феромонів в ACO, яка може відповідати або не відповідати сучасній та майбутній динаміці рою.

Одна з можливостей полягає в інтелектуальному прогнозуванні траєкторій на основі моделей та гібридному підході до планування, який поєднує метаевристики з моделями нейронних мереж. Такий підхід дозволяє еволюцію проактивного планування від чисто реактивних (або квазістатичних) форм, де рішення приймаються на основі можливих майбутніх станів сусідів. Покращення, що зумовлені цією додатковою функцією, включають не лише поточні вимірювальні вхідні дані планувальника, але й інформацію про прогнозовану траєкторію, що значно знижує ризики конфліктних ситуацій, що виникають раптово без будь-яких попередніх ознак.

Таким чином, сучасна тенденція в інтелектуальних алгоритмах планування для роїв БПЛА полягає в розробці гібридних систем, де метаевристичні методи

оптимізації поєднуються з моделями машинного навчання, що прогнозують поведінку навколишнього середовища. Це один із підходів, за допомогою якого можна досягти балансу між обчислювальною ефективністю та масштабованістю разом із відповідним рівнем безпеки в групових польотах; отже, це виправдовує подальший розгляд методів нейронних мереж для прогнозування траєкторії, як це передбачено в наступному розділі.

1.4. Нейромережі для прогнозування траєкторій

Як пояснювалося вище, для проактивного уникнення зіткнень необхідна здатність точно оцінювати майбутні шляхи сусідів. Цю проблему вирішують інструменти глибокого навчання, зокрема рекурентні нейронні мережі (RNN) та інші архітектури, здатні моделювати часові залежності.[18] RNN спеціально розроблені для роботи з послідовностями: вони містять петлі зворотного зв'язку, що дозволяє їм зберігати інформацію про попередні елементи в прихованому стані.[18] У завданні прогнозування траєкторії RNN може обробляти послідовні вимірювання положення та швидкості БПЛА, поступово «запам'ятовуючи» моделі руху. Однак базові RNN мають коротку пам'ять; зі збільшенням довжини послідовності чутливість до попередніх даних зменшується через ефект градієнтного спаду[18]. Отже, стандартний тип не може легко враховувати довгострокову залежність (наприклад, тенденцію до маневру).

LSTM –це архітектура RNN, що містить спеціальні комірки пам'яті з механізмом керування потоком інформації або "затворами" (вхідний затвор, затвор забуття та вихідний затвор)[18]. Функція вибіркового запам'ятовування та забування долає проблему градієнтного спаду у стандартних RNN. LSTM дуже добре справляються із завданнями, де контекстна інформація з попередніх точок даних у послідовності важлива для прогнозування, наприклад, аналіз часових рядів; обробка мовлення; машинний переклад тощо.[6] У роях БПЛА довгострокові тенденції (дрони поступово входять у поворот) можуть бути фіксовані за допомогою LSTM, що підвищує точність прогнозування траєкторії на великі відстані, однак вони є складнішими (мають більшу кількість параметрів) та

повільнішими у навчанні порівняно з простими архітектурами RNN[18]. Їх послідовний характер обмежує паралелізацію, що стає вузьким місцем у програмах реального часу.

Стробовані рекурентні одиниці (GRU) –це спрощена версія LSTM, що поєднує деякі вентилі для зменшення загальної складності [18]. У GRU є лише два вентилі –вентиль оновлення та вентиль скидання. Вентиль оновлення вирішує, яку частину нової інформації слід додати до стану, тоді як вентиль скидання вирішує, яку частину з попереднього стану потрібно забути. Оскільки кількість задіяних параметрів менша, GRU навчаються легше та швидше, ніж LSTM, не поступаючись їм у точності прогнозування [18]. Як зазначають дослідники, «GRU забезпечують кращу обчислювальну ефективність без значної втрати якості моделювання послідовностей» [6,19]. Таким чином, вони стають привабливими для використання в бортових комп'ютерах дронів, де ресурси обмежені. [6] Зазначається, що «модель GRU обчислювально дешевша, ніж LSTM, але зберігає здатність виявляти довгострокові залежності рухів БПЛА». Таким чином, запропонована архітектура повинна включати модель Bi-GRU з механізмом уваги для прогнозування нелінійних траєкторій сусідів. Двонаправлений GRU аналізує послідовність як у прямому, так і у зворотному напрямках, тим самим фіксуючи кращу контекстну інформацію, пов'язану з рухом, тоді як механізм уваги фіксує/фокусується на значних нещодавніх змінах (наприклад, раптовому маневрі), виявлених автоматично самою моделлю. Це призводить до точнішого та швидшого прогнозування траєкторії кожного сусіднього дрона, що доведено експериментальною точністю, досягнутою Bi-GRU з моделлю уваги [6].

Трансформери –це досить новий клас архітектур нейронних мереж, які докорінно змінили спосіб обробки послідовностей. Таким чином, довгі послідовності можуть бути змодельовані трансформерами без втрати будь-якої інформації про ранні кроки в контексті, оскільки «мережа бачить» весь контекст безпосередньо [18]. Ще одна велика перевага полягає в повністю паралелізованій обробці, що робить навчання дуже швидким на великих наборах даних [18]. В

останні роки моделі на основі трансформерів (GPT, BERT тощо) досягли найсучаснішої точності для багатьох завдань NLP, і вони поступово починають входити також у сферу робототехніки. Трансформери можуть бути застосовані як альтернативний підхід до прогнозування траєкторії до RNN у контексті роїв дронів: наприклад, модель трансформера траєкторії розглядає повну послідовність минулих станів для побудови прогнозів щодо позицій агентів, тим самим усуваючи проблеми довгострокової залежності [18]. Глибоке паралельне контекстне покриття досягається за допомогою трансформерів –зв'язки між усіма точками траєкторії розглядаються одночасно; отже, складні схеми маневрів потенційно можуть бути захоплені в них[19]. Однак вони мають досить багато параметрів і, отже, високі вимоги як до обчислювальних ресурсів, так і до обсягу даних під час їх навчання. Хоча модель RNN/GRU може легко перенавчатися на невеликому наборі (скажімо, менше 100) траєкторій, трансформаторам потрібні сотні тисяч, перш ніж вони доведуть будь-яку перевагу, окрім того, що обробка довгих послідовностей вимагає $O(n^2)$ пам'яті, що може стати проблематичним на повітряних системах.

У таблиці 1.3 підсумовано основні характеристики цих архітектур.

Таблиця 1.3

Порівняння нейромережевих архітектур для прогнозування траєкторій

Архітектура	Особливості	Переваги	Недоліки
1	2	3	4
RNN	Рекурентні зв’язки, простий блок пам’яті	Проста реалізація; швидке навчання на коротких послідовностях	Проблеми зі збереженням довгострокової пам’яті (згасання градієнта) [18]; послідовне опрацювання (погана паралелізація)

Продовження таблиці 1.3

1	2	3	4
LSTM	Комірка пам’яті з воротами (forget, input, output)	Утримує довгострокові залежності; вирішує проблему згасання градієнта [18]; високоточний контекстний аналіз	Багато параметрів (повільніше навчання) [18]; велике споживання пам’яті; все ще обмежена паралельність обчислень
GRU	Спрощена пам’ять з двома воротами (update, reset)	Менше параметрів – вище швидкість навчання [18]; майже LSTM-якість прогнозів при меншій складності [6, 18]	Може трохи поступатися LSTM в нюансах дуже довгих залежностей; послідовний характер обчислень зберігається
Transformer	Самоувага, відсутність рекурентності; паралельна обробка послідовності	Ефективно охоплює довгі послідовності (глобальний контекст) [18]; повна паралелізація прискорює навчання; найвища точність на великих данах	Величезні обчислювальні витрати і пам’ять [18]; потребує великого набору даних для навчання; складність реалізації на вбудованих системах

У таблиці 1.3 чітко показано, що «легкі» моделі (RNN, GRU) є простими, швидкими та точними за простих умов, але починають давати збій у складних випадках через їхню нездатність повністю охоплювати довгі контексти. З іншого боку, трансформатор бачить всю траєкторію та може вловлювати дуже складні шаблони маневрів сусідів, але вимагає потужного обладнання для виконання або коли дані надсилаються назад на певну базову станцію для обробки. Практична компромісна реалізація часто містить Bi-RNN/GRU з механізмом уваги [6]. Модель враховує рух сусіда «з минулого в майбутнє» та навпаки, автоматично навчаючись вибирати ключові моменти (зміни прискорення, повороти тощо). Прогнозування траєкторії стає достатньо точним для планувальника шляху, щоб уникнути конфліктних точок, що було досягнуто в сучасних гібридних системах запобігання зіткненням.

1.4.1. Формалізація завдання прогнозування траєкторій

Завдання прогнозування можна виразити в математичній формі наступним чином. Нехай стан сусіднього БПЛА в момент часу *t* описується вектором x_t (наприклад, включаючи координати, швидкості та прискорення: x, y, z, vx, vy, vz, ax, ay, az). Модель прогнозування F отримує на свій вхід послідовність станів для N попередніх кроків часу: $X_t = [x_{t-N+1}, \ldots, x_t]$. На виході вона генерує прогнозовану послідовність для наступних K кроків: $\hat{Y}_t = [\hat{y}_{t+1}, \ldots, \hat{y}_{t+K}]$, де $\hat{y}_i$ - прогнозований вектор станів. Навчання спрямоване на мінімізацію різниці між прогнозованим $\hat{Y}_t$ та реальними станами Y_t (наприклад, за допомогою функції втрат середньоквадратичної помилки MSE). Сучасні підходи також прогнозують коваріаційну матрицю помилок для підвищення надійності. Невизначеність прогнозування можна оцінити, і безпечний маршрут можна спланувати з урахуванням цієї оцінки [13, 24].

1.5. Архітектури систем навігації БПЛА

Архітектуру навігаційної системи слід визначати як організацію обчислень, що забезпечують обмін інформацією між дронами для підтримки скоординованого руху рою та запобігання зіткненням. Архітектури можна класифікувати як централізовані, децентралізовані та гібридні [17, 20].

У централізованій архітектурі певний головний вузол (наприклад, наземна станція або провідний дрон) збирає інформацію від усіх агентів та розраховує для них керуючі дії. Це типово для моделей «лідер-слідувач»: один або кілька дронів призначаються лідерами, інші йдуть за ними [20]. Перевагою централізації є відносна простота координації –лідер може підтримувати бажану формацію, а послідовники просто зберігають позицію відносно лідера [20]. Централізоване керування добре працює у структурованих завданнях (наприклад, польоти у формі), де заздалегідь відомо, хто за ким слідує, і потрібна чітка ієрархія [20]. Однак виникає ще більший недолік цього підходу –він дуже чутливий до збоїв як на боці лідерів, так і на боці зв'язку. У разі втрати провідного дрона або каналу

зв'язку з ним, вся група може втратити контроль [20]. Крім того, централізовані рішення не масштабуються: зі зростанням кількості дронів зростає і навантаження лідера, що зрештою робить обмін даними вузьким місцем (затримки, перевантаження каналу). Все більше сучасних досліджень визнають, що повністю централізовані рої не є надійними для великої кількості агентів у непередбачуваному середовищі [8, 17].

Децентралізована архітектура базується на припущенні, що БПЛА приймають рішення незалежно один від одного, спираючись на свої датчики та епізодичну інформацію, отриману від сусідів. Централізованого керування немає –інтелект розподіляється між агентами [17, 20]. Консенсусні алгоритми представляють собою повністю децентралізований підхід, оскільки всі дрони поступово приходять до певного спільного рішення (наприклад, швидкості або інтервали можуть бути зрівняні) [20]. Інший приклад –це поведінкові моделі, де кожен дрон просто дотримується певних правил (не стикатися з іншими; триматися близько; летіти до цілі) –як зграя птахів. Основна перевага, яку забезпечує така організація, полягає в відмовостійкості та масштабованості: збій в одному конкретному дроні майже ніколи не впливає на інші, а нові додані агенти не створюють непропорційного навантаження на систему [17]. Децентралізований рій продовжуватиме працювати з перерваним зв'язком, оскільки кожен дрон поводиться максимально незалежно. Проблема полягає в тому, щоб зробити їхні дії узгодженими, без будь-якого центрального координатора. Це вимагає спеціальних алгоритмів, які забезпечують конвергенцію в певну безпечну конфігурацію –наприклад, підходи, засновані на однакових обмеженнях швидкості, але застосовані попарно між усіма сусідами та доповнені бар'єрними функціями для гарантій безпеки [8]. Такі алгоритми складніше як розробити, так і перевірити; однак ORCA, BVO та нещодавній прогрес, досягнутий у запобіганні зіткненням багатороботних систем, показують, що повністю автономний децентралізований плавальний комплекс може функціонувати без єдиного лідера [4,8].

Гібриди намагаються взяти найкраще з обох світів.[12, 17] Зазвичай це передбачає встановлення певної ієрархії або кластеризацію всередині рою: формуються групи, серед яких один член обирається лідером, але всі лідери також координують свої дії на основі консенсусу на іншому рівні, вище за них. Така багаторівнева схема частково розвантажує єдиний центр і усуває єдину точку відмови, зберігаючи при цьому певні централізовані структури для легшого управління. Наприклад, із десятків дронів загалом в одному рої дронів можна призначити три чотири головні дрони, які обмінюються планами один з одним і призначають завдання підлеглим у своєму кластері. Гібридні архітектури також можуть бути функціонально розділені: глобальне завдання (наприклад, розподіл цілей між дронами) вирішується централізовано, тоді як локальні завдання (уникнення зіткнень, утримання строю) децентралізуються кожним дроном. Саме такий підхід здебільшого застосовується на практиці; існує наземний контроль, який періодично коригує місію рою, але між сеансами зв'язку дрони літають автономно.

Перспективним є федеративне навчання в роях, яке можна розглядати як гібридну архітектуру з точки зору адаптації моделі, і його слід окремо згадати. Федеративне навчання підтримує всі дрони для спільного навчання загальної моделі, наприклад, прогнозування траєкторії централізованих даних нейронної мережі без централізації самих необроблених даних. Кожен дрон навчається на власних локальних даних, а потім надсилає оновлені ваги моделі на сервер або лідера, де вони усереднюються для отримання покращеної глобальної моделі, яка знову повертається агентам. Таким чином досягається ефект спільного навчання, але система залишається розподіленою. Таку схему можна назвати гібридною, вона має централізований елемент (агрегацію моделей), але функціонує децентралізовано. [12] запропонував децентралізовану систему навігації роїв з федеративними моделями прогнозування адаптації на основі алгоритму FedProx. Це забезпечило кращу точність прогнозування в гетерогенному середовищі даних

та покращило колективні можливості рою уникати зіткнень з новими перешкодами.

Тому під час впровадження систем навігації рою слід знаходити компроміс між легкістю керування та надійністю. Централізоване керування простіше, але також дуже вразливе; децентралізоване керування стабільне, але потребує складних алгоритмів координації. Завдяки гібридним та багаторівневим архітектурам можна досягти високої автономності рою з керованістю, тому вони є найбільш використовуваними в сучасних дослідницьких прототипах [12, 17].

Таблиця 1.4

Порівняння архітектур систем навігації роїв БПЛА

Критерій	Централізована	Децентралізована	Гібридна
Масштабованість	Низька	Висока	Середня/Висока
Відмовостійкість	Низька (єдина точка відмови)	Висока	Середня/Висока
Складність реалізації	Низька/Середня	Висока	Дуже висока
Витрати на комунікацію	Високі (концентрація трафіку)	Низькі (локальний обмін)	Середні
Гнучкість та адаптивність	Низька	Висока	Середня/Висока
Придатність для щільних роїв	Обмежена	Висока	Висока

1.5.1. Вимоги до комунікаційних протоколів

Децентралізовані та гібридні архітектури критично залежать від комунікаційних середовищ як з точки зору надійності, так і швидкості. Для роїв БПЛА вони варіюються від стандартних протоколів (Wi-Fi IEEE 802.11, Bluetooth Mesh, радіомодеми) до більш специфічних (сенсорні мережі, протоколи для MANET –мобільних спеціалізованих мереж). Ключові вимоги такі:

Низька затримка: для реактивного запобігання зіткненням затримка обміну даними між сусідами повинна бути дуже малою (десятки мілісекунд).

Висока пропускна здатність: Пропускна здатність є вирішальним фактором для забезпечення високої пропускної здатності. Повинна бути достатня пропускна

здатність для передачі відео- та радіолокаційних даних або параметрів моделей нейронних мереж.

Надійність у динамічному середовищі: Протокол повинен мати можливості для швидких змін топології, втрат та перешкод –усі характеристики типові, а також для мереж пакетної радіозв'язку.

Масштабованість: Протокол не повинен суттєво погіршуватися зі збільшенням кількості вузлів (розміру рою).

Енергоефективність: Мінімізація споживання енергії на зв'язок для автономних БПЛА є важливим аспектом.

Висновки до першого розділу та постановка проблеми

Розділ 1 містить огляд існуючих підходів до запобігання зіткненням під час польоту рою БПЛА. Розглянута класифікація включає: реактивні алгоритми (штучні потенційні поля, ORCA тощо) є швидкими завдяки суворо локальному сприйняттю, але починають ефективно ламатися у складних сценаріях, для яких вони не були чітко розроблені (проблеми локальних мінімумів, тупикові ситуації) [1, 3]. Проактивні алгоритми просуваються далі, включаючи прогнозування поведінки сусідів, що значною мірою забезпечує безпеку та плавність траєкторій рою, проактивно, тоді як прості реактивні схеми для конфігурацій з високою щільністю стають занадто консервативними [5, 6]. Однак цей підхід вимагає складніших обчислювальних рішень, а також якісних моделей руху, тому розробляються гібридні методи, що поєднують надійні вимірювання датчиків, наприклад, FMCW-радар, з інтелектуальним прогнозуванням траєкторії на основі машинного навчання та оптимізацією метаевристики, що базується на плануванні шляху [9, 12]. Зокрема, радари повертають стабільні дані про відстань та швидкість незалежно від умов, у яких вони використовуються [10], рекурентні нейронні мережі (Bi-GRU, LSTM) ретельно прогнозують складні маневри сусідніх дронів [6], а такі алгоритми, як ACO та PSO, знаходять безпечні траєкторії з урахуванням цих прогнозів повністю децентралізованим чином [5,14,15].

Ще однією важливою тенденцією є відмовостійкість, або здатність до самовідновлення, кожного дрона в межах повністю бортового циклу «сприйняття-прогнозування-планування-дія», що усуває будь-яку єдину точку відмови та покращує масштабованість. Рої можуть зростати без втрати ефективності, оскільки обчислення розподілені між агентами [8, 17]. Інтелектуальні протоколи обміну повідомленнями зберігають пропускну здатність, періодична синхронізація моделей через федеративне навчання дозволяє всім дронам оновлювати свої нейронні мережі прогнозування без передачі необроблених даних [12]. Сучасне рішення містить різні інтегровані технології: радари як надійні очі для рою; нейронні мережі як прогностичний мозок та колективні алгоритми як механізм керування скоординованим рухом.

Децентралізовано. Цього прагне література, і цього прагне розробник нелінійної динаміки, тому що реальні сусіди мають наміри –не просто позиції, які потрібно займати в просторі, на основі якихось централізованих обчислень. Наміри, які динамічно змінюватимуться, щойно вони виявлять мій намір! Безпечно –навіть безпечніше, ніж сучасні найсучасніші рішення для великих роїв БПЛА з мінімальними між собою відстанями, де проактивність стає необхідністю, а не опцією через високу щільність = низьку відстань = високий ризик? Так, усі чотири одночасно, будь ласка: повністю децентралізовано; стійко (до невизначеності/шуму); обізнано про динамічні наміри сусідніх БПЛА та безпечно (<вимірюється> дуже консервативно)<у надзвичайно щільних роях<де відстані між дронами мінімальні. Тому проблема, що розглядається в цій магістерській дисертації, визначається як розробка гібридної інтелектуальної системи запобігання зіткненням для роїв БПЛА на основі точного радіолокаційного сприйняття[10], прогнозування траєкторії нейронної мережі[6,12] та децентралізованого планування безпечного маршруту[5,8,15]. Запропонований метод та архітектура системи описані в наступному розділі (Розділ 2), а його апаратна та програмна реалізація та експериментальна перевірка описані в розділах 3-4 разом з оцінкою ефективності порівняно з традиційними підходами.

РОЗДІЛ 2
ТЕОРЕТИЧНЕ ОБҐРУНТУВАННЯ ТА МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ СИСТЕМИ УНИКНЕННЯ ЗІТКНЕНЬ

У даному розділі представлено комплексне теоретичне обґрунтування принципів побудови систем уникнення зіткнень для групових польотів БПЛА. Основну увагу приділено аналізу архітектури бортових РЛС як ключового засобу детекції об’єктів, а також розробці математичного апарату для моделювання динамічної взаємодії апаратів у групі. Запропонований підхід дозволяє інтегрувати фізичні параметри радіолокаційного виявлення з інтелектуальними алгоритмами прийняття рішень для забезпечення безпечного маневрування в автономному режимі.

Попередній аналіз довів, що найкращим доступним варіантом для запобігання зіткненням є інтелектуальне гібридне проактивно-реактивне радіолокаційне нейронне децентралізоване рішення [12]. Тому нехай цей підхід буде математично обґрунтовано в наступному підрозділі. Цей розділ містить: (2.1) спрощену динамічну модель руху БПЛА з реальними обмеженнями; (2.2) феноменологічну модель радара FMCW та алгоритми для оцінки станів сусідніх БПЛА на основі відносних вимірювань, що повертаються такими радарами; і найголовніше, (2.3) формалізацію як задачі оптимального керування - планування траєкторії - із застосуванням усіх необхідних обмежень безпеки. Запропонована формулювання дозволяє безпосередньо інтегрувати в одну й ту саму задачу оптимізації обидва прогнози сусідніх траєкторій разом з їхніми невизначеностями.



Рис. 2.1 Загальна архітектура запропонованого гібридного підходу до уникнення зіткнень

2.1. Математична модель руху БПЛА

Для планування траєкторії рою рекомендована модель повинна бути кінемато-динамічною формулюванням, достатньо детальною, щоб відображати фізичні обмеження БПЛА, але залишатися простою для ефективної оптимізації. Тому ми використовуємо та приймаємо як таку модель подвійний інтегратор точкової маси, що описує лише поступальний рух БПЛА, ігноруючи його обертальну динаміку [21]. Це типове спрощення в більшості задач планування, оскільки воно значно знижує обчислювальне навантаження практично без втрати точності, що проявляється на рівні траєкторії рою.

Формально, стан БПЛА у часі t визначається вектором:

$$Xi(t) = [pi(t)T, vi(t)T]T = [xi(t), yi(t), zi(t), vx,i(t), vy,i(t), vz,i(t)]T,$$

де $\mathbf{p}_i(t) \in \square^3$ –вектор положення, а $\mathbf{v}_i(t) \in \square^3$ –вектор швидкості. Вектором керування $\mathbf{U}_i(t)$ є прискорення:

$$Ui(t) = ai(t) = [ax, i(t), ay, i(t), az, i(t)]T. \quad (2.1)$$

Рівняння руху мають вигляд системи диференціальних рівнянь:

$$\begin{cases} p\dot{i}(t) = vi(t), \\ v\dot{i}(t) = ai(t). \end{cases} \quad (2.2)$$

В матричній формі це можна записати як $\dot{\mathbf{X}}_i(t) = \mathbf{A}\mathbf{X}_i(t) + \mathbf{B}\mathbf{U}_i(t)$, де матриці відповідають моделі подвійного інтегратора:

$$A = \begin{bmatrix} 0_{3\times3} & I_3 \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}. B = \begin{bmatrix} I_3 \\ 0_{3\times3} \end{bmatrix}.$$

Така модель відома як point-mass model для мультикоптера і широко використовується у траєкторному плануванні [21].

Обмеження динаміки: Реальний БПЛА має кінематичні та динамічні обмеження, які включаємо в модель:

1. Обмеження прискорення: максимальна доступна тяга/прискорення обмежені:

$$||ai(t)|| \leq amax. \quad (2.3)$$

Обмеження накладається на норму вектора прискорень, що дозволяє повніше використати колективну тягу дрона [21].

2. Обмеження швидкості:

$$||vi(t)|| \leq vmax. \quad (2.4)$$

Зазначені обмеження ($\$a_{\max}\$, \$v_{\max}\$$) вибираються на основі технічних характеристик платформи. Типові значення для класу малих БПЛА наведені в Таблиці 2.1.

Таблиця 2.1

Типові параметри динамічної моделі для класу малих БПЛА (на прикладі квадрокоптера Iris)

Параметр	Позначення	Типове значення	Опис
Максимальна швидкість	$v_{\max}$	15 м/с	Безпечна швидкість польоту в умовах рою.
Максимальне прискорення	$a_{\max}$	5 м/с²	Обмеження, що визначається тягою двигунів.
Мінімальна безпечна відстань	$d_{\min}$	2.0 м	Фізичний мінімум для уникнення контакту.
Критична відстань (бар'єр)	d_{critical}	3.0 –4.0 м	Захисний шар для компенсації невизначеності.
Горизонт планування	T_h	3 –5 с	Інтервал часу для проактивного планування.

Запропонована модель є компромісом між точністю та простотою: вона враховує основні фізичні обмеження (інерцію, швидкість та обмеження прискорення) без ускладнень повної нелінійної аеродинамічної моделі [22]. Це призводить до фізично правдоподібних траєкторій, які також піддаються швидкому обчисленню.

2.2. Модель вимірювань бортового FMCW радара

Основним датчиком взаємного виявлення та параметрів руху між сусідніми БПЛА є бортовий радар міліметрового діапазону з частотно-модульованою безперервною хвилею (FMCW). Серед його переваг –всепогодність та пряме вимірювання відносної швидкості до цілі (ефект Доплера) [23, 24].

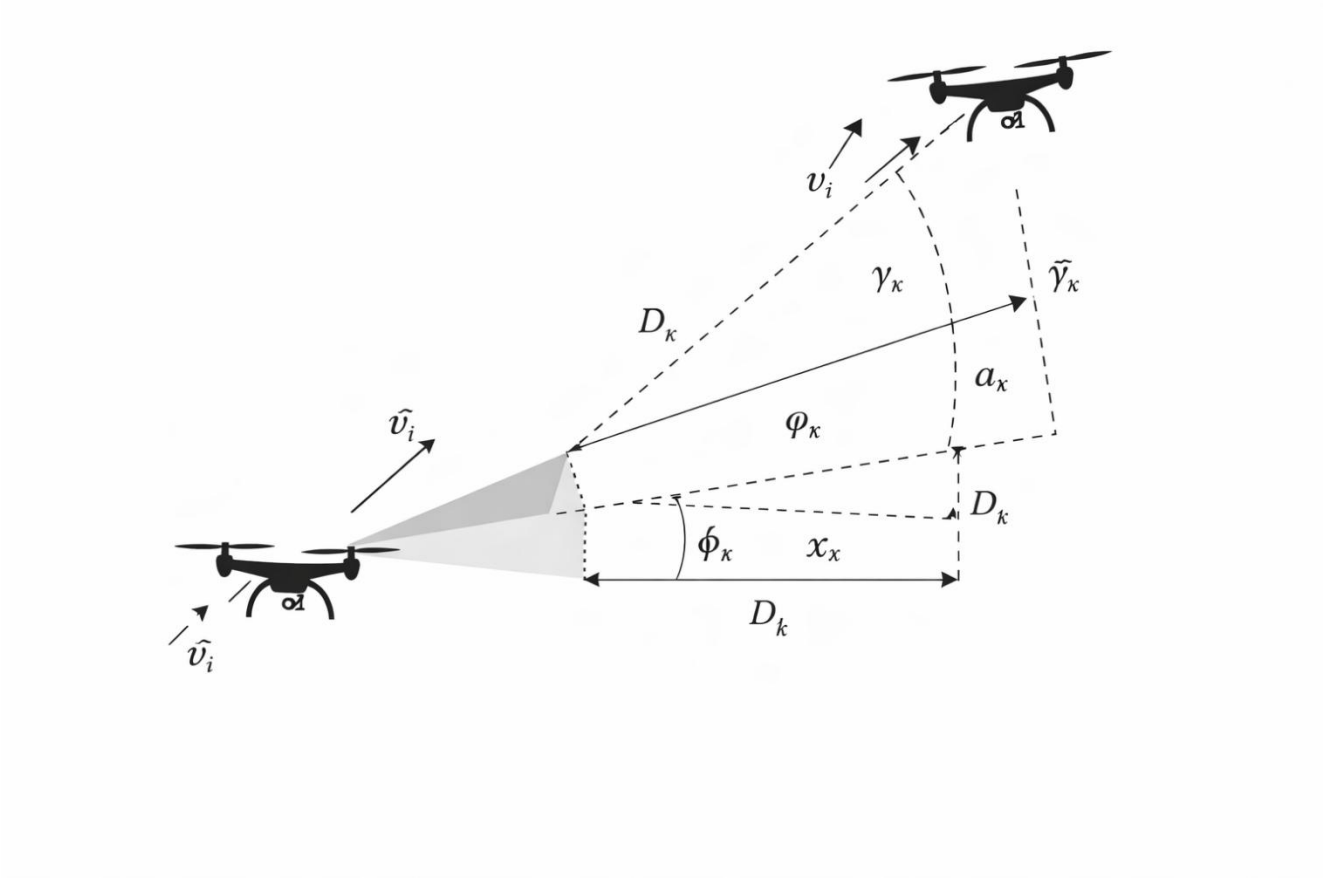


Рис. 2.2 Геометрія вимірювань бортового FMCW радара для взаємного виявлення БПЛА

Кожна детекція цілі надає вектор вимірювань $\mathbf{z}_k$\$:

$$zk=[\rho k,\theta k,\phi k,\rho \dot{k}]T, \tag{2.5}$$

де:

- ρ –дальність до цілі,
- θ –азимутальний кут,
- ϕ –кут місця,
- (\$$d_{\text{critical}}$\$), –радіальна швидкість.

Типові характеристики точності сучасних компактних FMCW радарів для БПЛА зведені в Таблиці 2.2.

Таблиця 2.2

Типові характеристики точності FMCW радара для БПЛА

Вимірюваний параметр	Позначення	Типова точність (1σ)	Примітка
Дальність	ρ	0.1 –0.2 м	Залежить від ширини смуги сигналу.
Азимутальний кут	θ	1 –3°	Визначається діаграмою спрямованості антени.
Кут місця	ϕ	1 –3°	Визначається діаграмою спрямованості антени.
Радіальна швидкість	$\dot{\rho}$	0.05 –0.1 м/с	Визначається доплерівською роздільною здатністю.

Модель спостереження, що зв'язує стан сусіда j з вимірами агенту i, має вигляд:

$$z_k = h(s_j) + v_k = \begin{bmatrix} r \\ \theta \\ \phi \\ \dot{r} \end{bmatrix} + v_k \qquad (2.6)$$

Тут $w_\rho, w_\theta, w_\phi, w_{\dot{\rho}}$ –гауссові шуми вимірювань з нульовим середнім та дисперсіями, зазначеними в Таблиці 2.2.

Оцінювання станів сусідів: Для обробки послідовності вимірювань $\mathbf{z}_k$ та отримання оцінок стану $\mathbf{X}_j(t)$ і коваріацій $\mathbf{P}_j(t)$ використовується розширений фільтр Калмана (EKF) [24]. Спрощена блок-схема цього процесу наведена на Рисунку 2.3.

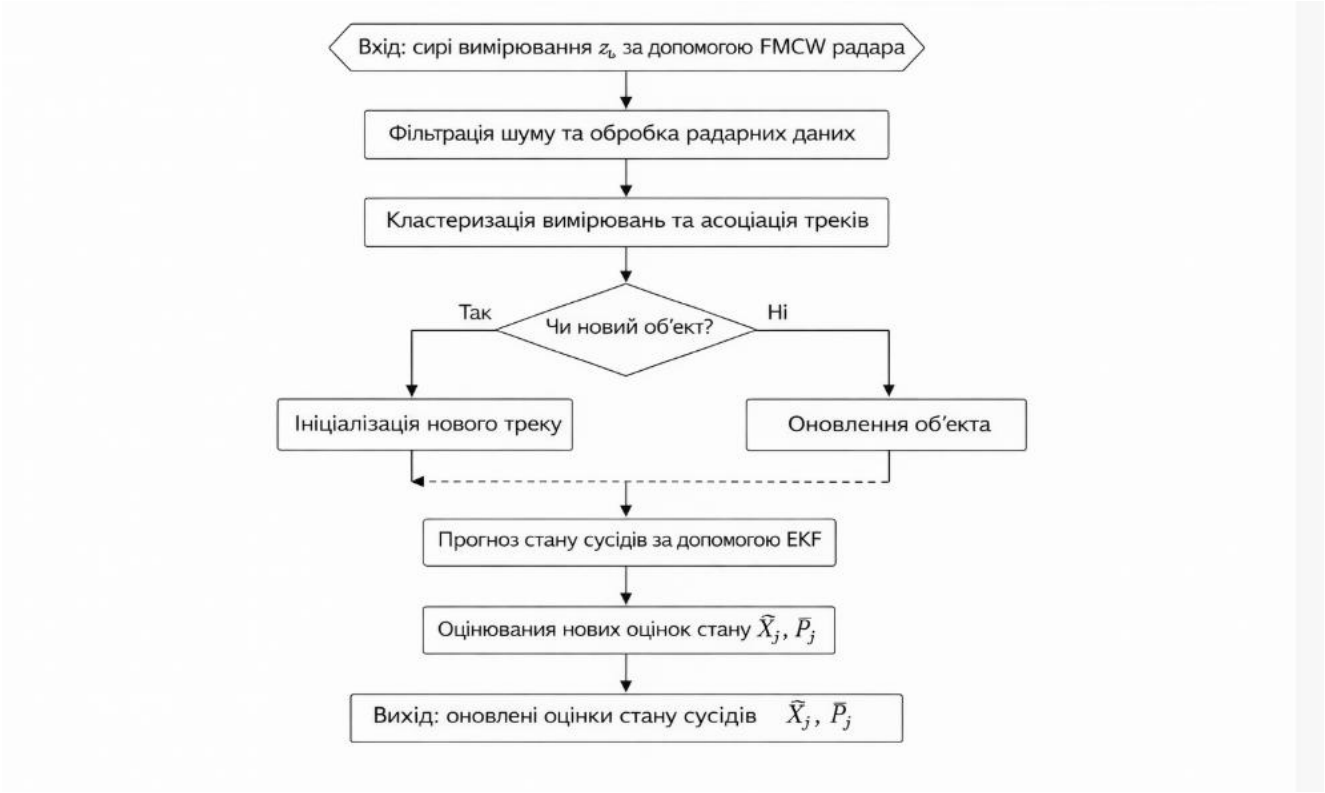


Рис. 2.3 –Блок-схема алгоритму обробки радарних даних та оцінювання стану сусідів

Таким чином, на борту кожного БПЛА формується структурована інформація про сусідів: $\mathbf{X}_j, \mathbf{P}_j$ для всіх виявлених агентів j.

2.3. Формальна постановка задачі оптимізації з обмеженнями

Поставимо задачу уникнення зіткнень як задачу оптимального керування на скінченному горизонті T_h, враховуючи жорсткі обмеження безпеки та прогноз поведінки сусідів. Кожен агент i на кожному кроці планування мінімізує цільовий функціонал J_i

Цільова функція J_i складається з кількох компонент:

$$J_i = J_{term} + J_{control} + J_{collision}. . \tag{2.7}$$

Термінальний штраф: Відстань до бажаної цільової точки $\mathbf{p}_{\text{goal},i}$ наприкінці горизонту.

$$J_{term} = w_1 \cdot \| p_i(T_h) - p_{goal,i} \|^2 . \tag{2.8}$$

Інтегральні штрафи за керування та відхилення: Відображають витрати енергії та прагнення до плавності руху.

$$J_{control} = \int_0^{T_h} [w_2 | \, a_i(t) | \, |^2 + w_3 \delta_i(t)^2] dt, \tag{2.9}$$

де $\delta_i(t)$ –міра відхилення від бажаної траєкторії або формації.

Штраф за наближення до інших агентів (проактивна безпека): Враховує прогнозовані траєкторії сусідів $\mathbf{p}_j(t)$.

$$J_{collision} = w_4 \int_0^{T_h} \sum_{j \neq i} f(\Delta_{ij}(t)) dt, \tag{2.10}$$

де $\Delta_{ij}(t) = \| \mathbf{p}_i(t) - \mathbf{p}_j(t) \|$ –відстань до прогнозованого положення сусіда j, а $f(\Delta)$ –бар'єрна функція (Рис. 2.4):

$$f(\Delta) = \begin{cases} exp(\alpha(d_{min} - \Delta)), якщо \, \Delta < dmin \\ 0, інакше. \end{cases} \tag{2.11}$$

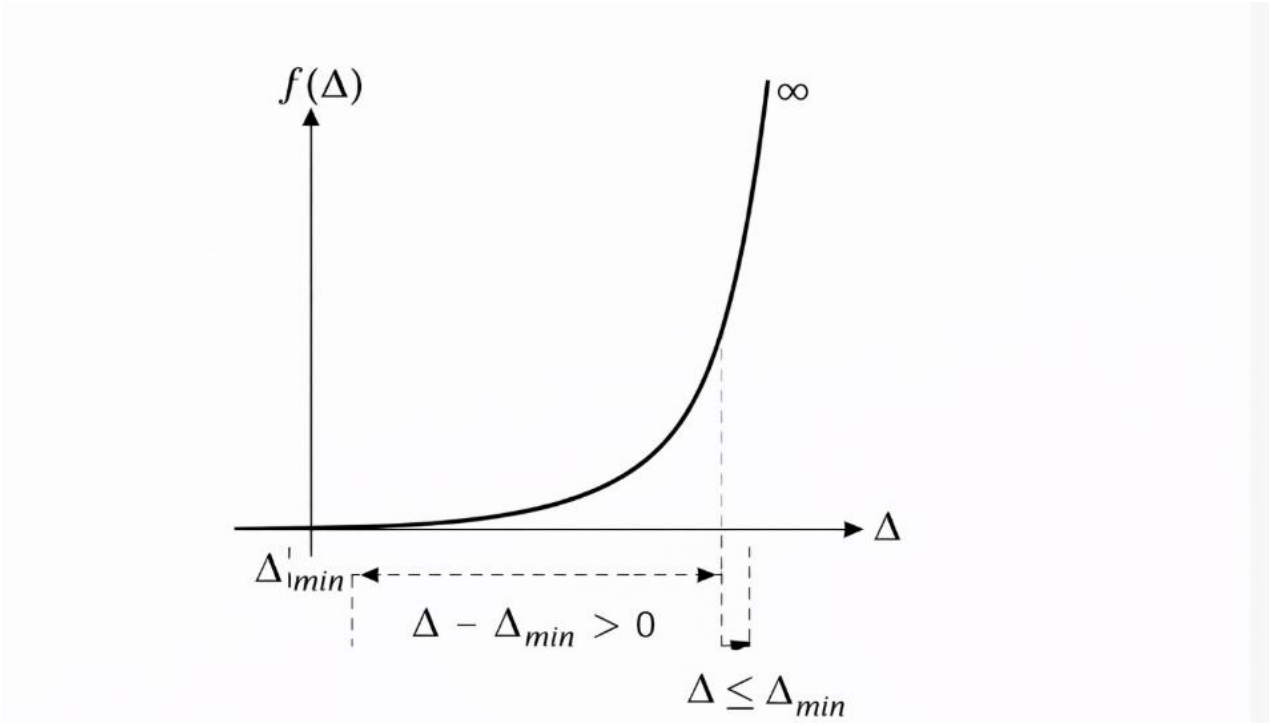


Рис. 2.4 Графік бар'єрної функції штрафу $f(\Delta)$ за наближення

Обмеження при мінімізації J_i:

1. **Динаміка руху:** Траєкторія повинна підкорятися рівнянням моделі (2.1) [22].

2. **Обмеження на керування та стан:** Виконуються нерівності (2.2) та (2.3).

3. **Жорстке обмеження безпеки (детермінований бар'єр):** Для всіх сусідів j і часу t:

$$|| p_i(t) - p^{?}_{j}(t) ||= d_{critical}, d_{critical} > d_{min}. \quad (2.12)$$

Це створює консервативний захисний шар навколо прогнозованої траєкторії сусіда [25].

4. **Ймовірнісне обмеження безпеки (стохастичний бар'єр):** Враховує невизначеність прогнозу $\mathbf{P}_j(t)$. Потрібно гарантувати, що ймовірність зближення нижче мінімальної відстані мала:

$$P(| \; p_i(t) - p_j(t)| \; |< d_{min}) \le \delta, \quad (2.13)$$

де δ –малий допустимий рівень ризику (напр., 0.01). Це обмеження можна апроксимувати вимогою неперетину еліпсоїдів невизначеності 3σ станів агентів [25, 26].

Таблиця 2.3а

Класифікація обмежень у постановці задачі оптимізації (Частина 1)

Тип обмеження	Математичний запис	Призначення
Рівність (динаміка)	$\dot{\mathbf{X}}_i = \mathbf{A}\mathbf{X}_i + \mathbf{B}\mathbf{U}_i$	Зв'язок стану та керування.
Нерівність (фізичні межі)	$\|\mathbf{a}i\| \le a\max\$,\$ \|\mathbf{v}i\| \le v\max$	Врахування можливостей БПЛА.
Детермінований бар'єр	$\|\mathbf{p}_i - \mathbf{p}j\| \ge d_{\text{critical}}$	Гарантія мінімального просторового розділення.
Стохастичний бар'єр	$\mathrm{P}(\|\mathbf{p}_i - \mathbf{p}j\| < d\min) \le \delta$	Урахування невизначеності з заданим ризиком.

Таблиця 2.3б

Характер обмежень задачі оптимізації (Частина 2)

Тип обмеження	Характер	Фізична інтерпретація
Рівність (динаміка)	Жорстке	Закон руху, що має виконуватися точно.
Нерівність (фізичні межі)	Жорстке	Технологічні межі швидкості та прискорення.
Детермінований бар'єр	Жорстке	Консервативна "буферна зона" навколо прогнозу.
Стохастичний бар'єр	М'яке (ймовірнісне)	Допустимий рівень ризику зіткнення (δ).

Сформульована задача оптимізації фактично є задачею оптимального керування зі змішаними обмеженнями. Її розв'язок забезпечує оптимальну траєкторію керування $\mathbf{U}_i^{(t)}$, а також траєкторії станів $\mathbf{X}_i^{(t)}$. У нашій системі децентралізований наближений пошук розв'язку реалізовано з використанням та застосуванням дещо модифікованої версії метаевристичного алгоритму (розділ 3).

Це пряма інтеграція прогнозу в критерій якості ($J_{\text{collision}}$) та обмеження, що створює гібрид між проактивним та реактивним: агент намагатиметься уникнути ймовірних конфліктів у майбутньому через м'які обмеження та жорсткі бар'єри (d_{critical}), що гарантує реактивну безпеку. Це належить до дуже сучасної тенденції в багатоагентному плануванні зі стохастичним MPC, як запропоновано в [25, 26].

2.4. Загальна архітектура системи AI-RDR (Artificial Intelligence –Radar, Decentralized & Robust)

AI-RDR –це децентралізована платформа на основі радара та алгоритмів штучного інтелекту для безпечної навігації рою БПЛА. Основними принципами, реалізованими в цій системі, є: децентралізація (кожен агент працює незалежно без центрального контролера), адаптивність, що дозволяє агентам налаштовуватися відповідно до змін у середовищі, стійкість, що працює в режимі реального часу та залишається життєздатною навіть при часткових збоях, та постійний цикл обробки,

починаючи від збору даних датчиків до прогнозування руху об'єкта, потім планування шляху та, нарешті, надсилання команд керування, де всі модулі інтегровані в один конвеєр[20].

Чотири основні блоки загальної архітектури (див. текстовий опис блок-схеми):

- Модуль сприйняття –попередньо обробляє необроблені дані радіолокаційного спостереження FMCW, видаляє шум та кластеризує виявлення для отримання інформації про цілі. Містить алгоритми відстеження CFAR, DBSCAN, EKF (див. розділи 2.5.1—2.5.3). На його виході формується об'єктне представлення навколишнього простору (сітка заповненості або список цілей з параметрами).
- Модуль прогнозування –отримує вхідні структуровані дані про цілі (їхні координати, швидкості) від модуля сприйняття та прогнозує їхні майбутні траєкторії за допомогою рекурентної нейронної мережі (архітектура Attention-Bi-GRU, розділ 3.3). Він робить прогнози послідовності координат для кожної цілі разом з оцінками невизначеності (розбіжності між різними прогнозами).
- Модуль планування –отримує інформацію про майбутнє розташування сусідів від Модуля прогнозування та генерує безпечний маршрут для агента (БПЛА). Використовується модифікований алгоритм оптимізації колонії мурах у безперервному просторі, в який інтегровані евристики для врахування прогнозних траєкторій (розділ 3.4). Планувальник формує шлях з урахуванням перешкод, обмежень кривизни та заторів, використовуючи передані прогнози інших агентів.
- Модуль виконання/керування –перетворює розрахований маршрут та план маневру в команди безперервного керування для ARS (автопілота) БПЛА, контролює їх виконання та вносить корективи у разі відхилень.

Весь конвеєр обробки даних та команд працює циклічно: на кожному циклі агент отримує нові дані радіосприйняття, оновлює оцінку навколишнього середовища, прогнозує рух сусідів, планує оптимальний маршрут на наступні кілька секунд та виконує першу команду цього маршруту, після чого цикл

повторюється. Взаємодія між модулями також є циклічною: результати від модуля сприйняття передаються як вхід до модуля прогнозування, потім до модуля планування, і нарешті до виконавчого модуля. У свою чергу, модуль виконання повертає інформацію про фактичний стан (агента, наприклад, його власне положення та швидкість) на вхід сприйняття для наступного циклу, утворюючи замкнутий контур керування.

Кожен агент (БПЛА) запускає всі свої модулі незалежно, обмінюючись лише мінімально необхідною інформацією, наприклад, параметрами прогнозованих траєкторій, із сусідами через комунікаційну мережу для підтримки роботи в режимі реального часу з обмеженим зв'язком між агентами. Це робить систему відмовостійкою: збій на одному вузлі не призводить до виходу з ладу всієї системи. Також забезпечується детермінований час відгуку завдяки локальній обробці разом з автономністю агента [16], [34].

2.4.1. Взаємодія модулів системи

Поточна схема обробки даних у AI-RDR складається з таких послідовних кроків (див. умовну блок-схему нижче):

1. Крок 1 (сканування та виявлення): Радар FMCW формує необроблені радіолокаційні зображення навколишнього простору. Після завершення швидкого перетворення Фур'є (FFT) розраховується радіолокаційна діаграма –дальність, швидкість Доплера, азимут виявлених відбиттів. На цьому етапі застосовується адаптивна схема CFAR для фільтрації перешкод та шуму, а також алгоритм кластеризації DBSCAN, який кластеризує виявлені точки в цільові кластери (розділ 3.2.1). Це забезпечує початковий список цільових об'єктів у полі зору [31].

2. Крок 2 (відстеження цілі): Для кожного окремого кластера ініціюється один трекер на основі розширеного фільтра Калмана (EKF). Модель стану містить координати та швидкості цілі. Таким чином, шляхом обчислення EKF[78] вона оновлює положення та швидкості цілей з урахуванням шуму датчиків під час виявлення, а також динаміки руху, формуючи таким чином поточну оцінену траєкторію для кожної виявленої цілі.

3. Крок 3 (формування просторового представлення): Результати відстеження інтерпретуються як організоване представлення середовища (розділ 3.2.3). Це може бути сітка зайнятості –дискретна карта ймовірності зайнятості простору, або список об'єктів з параметрами (координати, розміри тощо). Кожна комірка (або об'єкт) представляє виявлені перешкоди або інші літальні апарати.

4. Крок 4 (прогнозування траєкторії): Майбутні траєкторії всіх видимих на даний момент цілей прогнозуються на основі сформованого представлення середовища та історичних спостережень (розділ 3.3). Для цього використовується нейронна мережа з архітектурою Attention-Bi-GRU. Вона приймає на вхід послідовності історичних станів для кожної цілі (їхні координати та швидкості за останній період) та виводить прогнозовані послідовності їх координат протягом прогнозованого горизонту. Механізм Attention допомагає вибрати найбільш релевантні часові кроки та цілі для досягнення високої точності прогнозування [11].

5. Крок 5 (планування шляху): Враховуючи поточне положення власного БПЛА, а також прогнозований рух сусідніх літальних апаратів (з кроку 4), цей модуль планування знаходить оптимальний безпечний маршрут, обговорений у розділі 3.4. Використовується модифікована модель оптимізації мурашиної колонії, в якій безперервний простір був адаптований для застосування евристики, яка штрафує маршрути через прогнозовані перешкоди. Виходом є послідовність координат траєкторії, наприклад, лінійні та поворотні сегменти, які мінімізують ймовірність зіткнень із сусідами та перешкодами[5][8].

6. Крок 6 (виконання та керування): БПЛА починає реалізацію початкових етапів запланованого маршруту. Команди подаються на автопілот або контролер руху (швидкість цілі, курс, висота), який керує двигуном та елементами керування. Паралельно здійснюється також моніторинг виконання маршруту: якщо під час руху датчики виявляють різку зміну ситуації (наприклад, на його шляху з'являється нова ціль), то ця система швидко ініціює перепланування маршруту (див. розділ 3.6.2) [32].

Таким чином, потік даних між модулями завжди безперервний та циклічний: кожен модуль надає результати як попередньому, так і наступному етапу. Обмін інформацією між агентами містить лише прогнозовані траєкторії або інші дані про обслуговування (для координації), без централізованого контролю.

2.4.2. Вимоги до децентралізованої роботи

Оскільки AI-RDR орієнтований на децентралізований рій агентів, система повинна відповідати таким основним вимогам:

- Детермінований час відгуку. Система працює в суворих рамках реального часу: між обробкою радіолокаційних даних та видачею керуючого сигналу має пройти не більше гарантованого інтервалу. Відсутність централізованого сервера дозволяє суворо обмежити затримку реагування на нову інформацію [15], [22].
- Автономність агента. БПЛА повинен мати здатність приймати рішення на основі своєї локальної інформації та тієї, яку він отримує від сусідів, тому кожен агент містить повний стек алгоритмів (перспективи, прогнозування, планування) і не залежить від одного вузла керування [4], [10].
- Комунікація відбувається обмеженим чином. Між агентами залишаються канали зв'язку, але вони обмежені ресурсами (навантаження мережі, можливі збої). Система також повинна працювати з ситуаціями одноразових збоїв повідомлень, в основному обмін інформацією між одним або кількома сусідами. Агенти мінімізують інформацію, яку вони надсилають (наприклад, зведену інформацію про траєкторії), щоб зберегти пропускну здатність [14],[20].
- Робота в реальному часі. Алгоритми обробки даних та навчання повинні бути здатні працювати на частоті, що перевищує критичний період відгуку, для досягнення роботи в реальному часі. Наприклад, вони повинні виконувати фільтрацію та кластеризацію радіолокаційного сигналу за час, коротший за інтервал радіолокаційного вимірювання, а також виконувати, коли модуль планування видає новий план на горизонті тривалістю приблизно 5-10 секунд [20][34].

- Робастність. Система повинна бути стійкою до помилок в окремих компонентах: втрата зв'язку з деякими агентами або вихід з ладу їхніх датчиків не повинні спричиняти колапс усієї системи. Це передбачає використання механізмів перевірки вихідних даних, багатий набір евристик планування та можливість перепланування у разі виникнення надзвичайних ситуацій [15], [26].

Децентралізоване управління також робить систему стійкою до деяких непередбачених ситуацій та здатною функціонувати як компонент величезного рою безпілотників з обмеженим адмініструванням.

2.5. Модуль обробки даних FMCW-радара та оцінювання стану

FMCW-радар (частотно-модульований безперервний радар) визначається як тип радара, який передає сигнал безперервно на поступово змінних частотах. Вузький діапазон розрізнення дозволяє одночасно визначати як відносну швидкість, так і відстань до об'єкта від цілі за допомогою FMCW-радара.[19], [23] Загальне застосування в БПЛА включає виявлення перешкод та інших повітряних об'єктів на відстані до кількох сотень метрів. Його принцип роботи можна підсумувати таким чином: отриманий сигнал змішується з поточним передаваним сигналом, «гібридним» сигналом, і зсув частоти або частота зміщення має прямий зв'язок з відстанню до цілі, тоді як відфільтрований кут талії або крок має прямий зв'язок з її відносною швидкістю до/від радара. Цілі після FFT є тривимірними масивами даних, тобто (дальність, доплерівський кут, азимут).

2.5.1. Алгоритм фільтрації та кластеризації

1. До необроблених радіолокаційних даних застосовується тривимірне швидке перетворення Фур'є (3D FFT). Висновок можна зробити з того факту, що операція обробки сигналу, триразова вздовж осей часу, азимута та місця, формує радіолокаційний «об'єм» даних, що містить інформацію про амплітуду в кожній комірці відстані, швидкості та кута огляду.[31]

2. CFAR-фільтрація: алгоритм CFAR (постійна частота хибних тривог) працює на отриманому спектральному «знімку». Він обчислює адаптивний поріг для кожної комірки даних на основі сусідніх значень (локальний шум) та виявляє ті комірки, де сигнал перевищує цей поріг (ймовірна присутність цілі). У CFAR використовуються різні стратегії, наприклад, CA-CFAR, OS-CFAR –усі вони порівнюють амплітуду отриманого сигналу з локальним порогом, що мінімізує помилки за наявності гетерогенних шумових перешкод.[31]

3. Кластеризація точок: CFAR –це розріджений набір виявлень. Ці точки, що належать до однієї фізичної цілі, групуються за допомогою кластеризації. Наприклад, DBSCAN[20] об'єднує виявлення за критерієм щільності, що призводить до окремих кластерів –сферичного розподілу виявлень, пов'язаних з одним об'єктом (лише ті виявлення, де CFAR перевищує поріг, потрапляють до кластерів). Це надає список кластерів (перешкод/літаків) з координатами (центр мас кластера в 3D-радарній системі) та проміжками.

4. Післяобробка та вилучення центру кластера: центроїд або усереднені параметри кластера містять приблизне поточне положення цілі. Потім вони передаються на етап відстеження.

Наступний конвеєр обробки демонструє хорошу ефективність виявлення цілей: (FFT → CFAR → DBSCAN). Як зазначали Камана та ін. (2020), ця послідовність значно покращує виявлення об'єктів: Після OS-CFAR точки групуються за допомогою DBSCAN, що дозволяє «підсвітити» наявні об'єкти [7].

Псевдокод може бути таким для обробки одного радіолокаційного кадру спрощеним способом, розміщений в додатку А.

Фільтрація CFAR та кластеризація DBSCAN, два процеси, що гарантують, що шумові сплески не дають хибних виявлень,[12] також відіграють певну роль у забезпеченні об'єднання відповідних цільових елементів в одну сутність. Тому на цьому етапі вже існує приблизний чистий список цілей з їх положеннями та розмірами.

2.5.2. Застосування розширеного Калманівського фільтра (EKF)

Після кластеризації починається відстеження –оцінка та відстеження стану кожної виявленої цілі з плином часу. Для цієї мети використовується розширений фільтр Калмана (EKF), оскільки система не є повністю лінійною; тобто як рух об'єкта, так і радіолокаційні вимірювання містять деякі елементи нелінійності.

Вектор стану зазвичай містить координати об'єкта в просторі (x, y, можливо, z для 3D) та їх компоненти швидкості. Наприклад, для площинного руху ми маємо вектор стану $x = [x, y, v_x, v_y]^T$.

Процеси динаміки руху описуються функцією f(x,u), де u –зовнішній вхідний сигнал (може бути відсутнім для інерційних цілей). Модель вимірювання h(x) описує те, що ми отримуємо на вхід від датчика: наприклад, радар безпосередньо вимірює відстань r, кут θ та, можливо, доплерівську складову v_r. Ці моделі задаються рівняннями (для неперервної системи та дискретних вимірювань):

$$x\dot{(t)} = f\left(x(t), u(t)\right) + w(t), z_k = h\left(x_k\right) + v_k\,, \tag{2.14}$$

де w(t) та v_k –це гаусівські вектори шуму процесу та вимірювання з відомими коваріаціями Q та R. Для дискретного кроку отримуємо:

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}) \tag{2.15}$$

$$P_{k|k-1} = F_{k-1} P_{k-1|k-1} F_{k-1}^{\bullet} + Q_{k-1}\,, \tag{2.16}$$

де $F_{k-1} = \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}_{k-1|k-1}}$ –матриця Якобі для лінеаризації. Після цього виконується корекція (оновлення) на основі нового вимірювання z_k:

$$K_k = P_{k|k-1} H_k^{\bullet} \left(H_k P_{k|k-1} H_k^{\bullet} + R_k\right)^{-1} \tag{2.17}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \left(z_k - h(\hat{x}_{k|k-1})\right) \tag{2.18}$$

$$P_{k|k} = \left(I - K_k H_k\right) P_{k|k-1}\,, \tag{2.19}$$

де $H_k = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}_{k|k-1}}$ –матриця Якобі функції вимірювання, а K_k –матриця Калмана.

Цей цикл «прогнозування-оновлення» повторюється для кожного нового вимірювання [24]. Таким чином, EKF дає вектор оцінки x^_(k|k) стану цілі та коваріацію помилки P_(k|k). Він знаходить застосування в задачах відстеження, оскільки видаляє випадковий шум, забезпечує прогнозування руху навіть тоді, коли миттєві вимірювання недоступні. Водночас зазначимо невдачі EKF –якщо моделі руху або вимірювання неточні або початкові умови встановлені неправильно, фільтр не збігається правильно. Наприклад, у випадку сильної нелінійності або великої перешкоди оцінка може «розходитися». Зокрема, якщо фактичний рух об'єкта різко відхиляється від передбачуваної моделі (наприклад, раптові маневри не враховані), з EKF отримується дуже зміщена оцінка, і тому потрібен перезапуск або повторна ініціалізація відстеження [12],[24].

2.5.3. Формування просторового оточення

Результати відстеження (списки об'єктів з їх положеннями та швидкостями) слід компілювати в такі форми даних, які б полегшували подальшу обробку нейронною мережею, а також планувальником. Загальним типовим форматом є сітка зайнятості (ймовірнісна карта зайнятості простору) або, можливо, просто спрощений список об'єктів. Карта сітки зайнятості (OGM) - це дискретна сітка - 2D або 3D - де кожна комірка містить інформацію про ймовірність зайнятості/вільності/незайнятості. Планувальник перевіряє зіткнення, використовуючи цю карту: комірки з високою ймовірністю зайнятості позначені як небезпечні. Це ілюструє, як трекер розміщує радарні відображення в комірках карти: комірки об'єктів отримують "1", вільні - "0", невизначеність=0,5 тощо; таке відображення може оновлюватися в режимі реального часу (байєсівське, TSDF тощо). Оскільки використовується радар FMCW, швидкість також може

враховуватися: деякі підходи доповнюють OGM "динамічними" членами (додатковий вектор швидкості в комірці), розділяючи рухомі об'єкти [21].

Інший варіант –створити список об'єктів. У цьому випадку кожна ціль передається з параметрами (тип об'єкта, координати центру, радіус/розміри, швидкість руху тощо). Таке представлення більш безпосередньо використовується тими мережами, які можуть працювати з набором об'єктів. Наприклад, модуль прогнозування в розділі 3.3 може послідовно приймати параметри цілей, виявлених трекером.

Загалом, просторове представлення, яке будується трекером, можна описати як операцію, яка перетворює простий невпорядкований список виявлених точок на деякі структуровані дані –карту комірок або, можливо, просто набір записів, що містять координати об'єкта. Це полегшує подальшу генерацію вхідних даних для прогностичної нейронної мережі та інших модулів усередині цієї системи.

2.6. Модуль прогнозування траєкторій на основі нейромережі Attention-Bi-GRU

Для прогнозування траєкторій прогнозованих об'єктів було обрано архітектуру, що поєднує Bi-GRU та Attention. Bi-GRU дозволяє обробляти часові послідовності в обох напрямках (пряма та зворотна історія), витягуючи контекст з усього вхідного кадру. Механізм уваги призначає вагові коефіцієнти важливості на кожному кроці часу для кожної цілі, дозволяючи моделям зосередитися на більш релевантних даних. Подібні підходи досягли чудових результатів у завданнях прогнозування траєкторій, що включають складні рухи; наприклад, гібрид Bi-LSTM та Bi-GRU з механізмом уваги, запропонованим для прогнозування руху суден, значно перевершив інші методи [11]. Саме це виправдовує вибір Bi-GRU (який ефективно збирає інформацію з обох напрямків) разом з увагою для адаптивного вибору корисної інформації. Це рекурентна архітектура, поєднана з увагою, яка дозволяє моделі ефективно прогнозувати складні відхиляючі траєкторії в реальних умовах неповного спостереження [11].

2.6.1. Архітектура мережі

Структура мережі визначається наступним чином:

- Вхідний шар отримує дані як послідовність попередніх станів цілей. Наприклад, для кожної цілі формується послідовність $\left[x_{t-N}, y_{t-N}, v_{x,t-N}, v_{y,t-N}, \ldots, x_t, y_t, v_{x,t}, v_{y,t}\right]$ довжиною N+1 (останній поточний час). У випадку, якщо прогнозується більше однієї цілі одночасно, одним із варіантів може бути надання їх як окремих послідовностей паралельним Bi-GRU (або одну за одною з унікальними маркерами).
- Шари Bi-GRU. Шари захоплення (замкнуті рекурентні блоки) обробляють послідовність у двох напрямках –вперед і назад, генеруючи приховані представлення $h_t^{(f)}$ та $h_t^{(b)}$
- . Двонаправлена обробка дозволяє враховувати кореляції у двох напрямках часу. Кількість шарів та розмір стану вибираються експериментально.
- На виході Bi-GRU для кожного кроку часу ми маємо приховані вектори станів. Механізм уваги обчислює ваги важливості для кожної цілі і на кожному кроці часу t. Наприклад, ми можемо визначити оцінку сумісності між поточним станом та прихованим вектором: $e_{t,i} = \text{score}(h_t, u_i)$, яка потім нормалізується за допомогою softmax для отримання $\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_j \exp(e_{t,j})}$. Вектор контексту стає зваженою сумою сусідніх прихованих станів: $c_t = \sum_i \alpha_{t,i} h_i$. Цей контекст або набір векторів контексту переноситься вперед. Механізм уваги дозволяє моделі зосереджуватися на певних цілях або фрагментах історії як найбільш релевантних під час прогнозування [11].
- Вихідні шари. Контекстні вектори з шарів уваги об'єднуються в щільний шар для створення кінцевих прогнозованих змін стану. Для кожної цілі мережа створює послідовність майбутніх координат $\left[x_{t+1}, y_{t+1}, \ldots, x_{t+H}, y_{t+H}\right]$,

де H –горизонт прогнозу (на кілька секунд вперед). Вона також може забезпечити оцінку невизначеності: наприклад, окрім простого надання середнього значення координат, модель можна навчити так, щоб вона надавала дисперсію або параметри гауссового розподілу (гіпотетична MSE або логарифмічні втрати)[12].

2.6.2. Підготовка даних та навчання

Навчальний набір будується з реальних історичних даних траєкторії та/або синтетичних симуляцій. Для кожного сегмента спостережень створюється вхідна послідовність (історія довжиною N) разом з відповідним вихідним даними, що є справжньою траєкторією для H кроків вперед. Функція втрат зазвичай визначається як середньоквадратична помилка (MSE) між прогнозованими координатами та реальними траєкторіями цілей. Оптимізатор, наприклад, Adam або RMSProp, оновлює ваги мережі на основі градієнта помилки.

Для оцінки якості використовуються метрики помилок траєкторії, наприклад, середня помилка зміщення (ADE) та кінцева помилка зміщення (FDE). Також слід перевірити помилку валідації, щоб модель не перенавчалася. Для запобігання перенавчанню застосовуються звичайні стандартні методи: регуляризація (L2), випадіння в рекурентних шарах, перехресна валідація та своєчасне раннє зупинення навчання, коли втрати валідації збільшуються [13], [35].

Таким чином, мережа навчається шукати узагальнені закономірності в минулому русі (припустимо, плавні криві або типові маневри цілей), і це дозволяє їй точно передбачати нові траєкторії.

2.6.3. Вихідні дані модуля

На виході модуля прогнозування ми маємо два компоненти:

1. Для кожної виявленої цілі існує послідовність прогнозованих точок траєкторії $\left[(\hat{x}_{t+1}, \hat{y}_{t+1}), \ldots, (\hat{x}_{t+H}, \hat{y}_{t+H})\right]$ на горизонті прогнозування H. Вони передаються модулю планування як локальні цілі для майбутніх часових кроків.

Оцінки невизначеності прогнозу. Кілька дуже сучасних архітектур дозволяють використовувати вторинний вихідний канал для оцінки достовірності

або дисперсії, пов'язаної з кожною прогнозованою координатою (наприклад, припускаючи параметри нормального розподілу),^1 або ж підтримують прогнозування за ансамблями, з яких можна обчислювати дисперсії, як у випадку використання відсіву під час тестування для отримання розкиду результатів.^2 Це дає планувальникам ще один спосіб: «м'яке резервування» простору навколо цілей на основі того, наскільки невизначені траєкторії прогнозу знаходяться далеко від їх найімовірніших місць розташування. Простішим реалізаціям потрібні лише середні значення μ та стандартні відхилення σ, щоб вони могли будувати довірчі інтервали за положеннями об'єктів,[12],[25]

Таким чином, модуль Attention-Bi-GRU надає планувальнику як прогнозовані майбутні позиції кожної цілі, так і міру, що вказує на їхню невизначеність. Цей інструмент використовує цю інформацію для прокладання безпечного шляху не лише для обходу фактичних перешкод, але й для уникнення ймовірних шляхів сусідніх агентів з належним урахуванням їхньої невизначеності.

2.7. Модифікований алгоритм мурашиних колоній (ACO) для планування безпечного шляху

Класична оптимізація колонії мурах (ACO) –це ймовірнісний метаевристичний алгоритм для задачі найкращого шляху в графі. Концепція лежить в основі поведінки реальних мурах: феромонні сліди дозволяють штучним «мурахам» вибирати короткі або хороші шляхи з високою ймовірністю, а отже, кращі рішення. Формально ACO визначається як алгоритм для шляхів у графі, де можливі рішення (шляхи) представлені як послідовності вузлів у графі; феромонні коефіцієнти та евристичні значення контролюють вибір наступного кроку. У дискретних задачах траєкторія складається з послідовних кроків між фіксованими вузлами.[5]

Класична схема ACO (дискретний випадок):

1. Крок 1. Ініціалізація: Встановлення початкових значень феромонів τ_ij на всіх ребрах графа.

2. Крок 2. Імітація мурахи: Кожна мураха починає з початкового вузла та рухається до цілі, на кожному кроці вибираючи наступний вузол j з ймовірністю, пропорційною $\tau_{ij}^{\alpha}\eta_{ij}^{\beta}$ (де τ_{ij} –кількість феромону на ребрі, η_{ij} –локальний евристичний коефіцієнт, α,β –ваги).

3. Оновлення феромонів: Кожна мураха після завершення свого маршруту кладе феромон на шлях відповідно до якості знайденого рішення. Потім відбувається випаровування феромонів (значення τ_ij зменшується на деякий фіксований коефіцієнт).

4. Повторення ітерацій: Кроки 2-3 повторюються багато разів, що дозволяє поступово накопичувати феромони на хороших шляхах. Зрештою, найкоротший шлях матиме максимальну привабливість для мурах.[5]

Алгоритм визначено над дискретною областю (спільні шляхи рою в графі). Його необхідно додатково визначити для неперервної області, як у нашому випадку руху дрона над площиною або в межах деякого об'єму.

2.7.1. Адаптація до безперервного простору

У неперервному середовищі траєкторію агента можна визначити як послідовність точок з неперервними координатами. Тому виникає потреба якимось чином зобразити можливі шляхи у вигляді графіка. Ми пропонуємо таке представлення:

- Представлення шляху: шлях кодується як послідовність контрольних точок у просторі. Для застосування ACO буде зручно розглядати ці точки як вузли графа. Граф може бути побудований на основі деякого розбиття простору (наприклад, регулярної сітки або адаптивного поділу на квадранти). Кожна вершина відповідає деякій координаті (x,y), а ребра є можливими відрізками ліній між близькими сусідами.
- Побудова графу пошуку: як варіант, можна створити дерево шляхів. Коренем буде початкова позиція агента, потім шар дочірніх вузлів є потенційними точками наступного кроку і так далі. Інший підхід –фіксована сітка: агент рухається

по краях цієї сітки. Зауважте, що в безперервному ACO мурахи не досліджують усі точки з однаковою частотою: часто використовується жорсткий поділ кроків (наприклад, кожен наступний вузол –це фіксований крок руху від поточного).

- Мурахи рухаються: штучні мурахи «відвідують» координати, подібно до дискретного випадку, але тепер координатами можуть бути будь-які точки в області. На ітерації кожна мураха вибирає наступну точку на графіку відповідно до ймовірностей, що залежать від феромону на відповідному сегменті та локальної евристики. Якщо використовується адаптивна сітка, то вибір обмежений вузлами, які походять з поточної точки. При використанні дерева мурахи можуть розгалужуватися або відкидати гілки [8],[9].

Зауважте, що представлення шляхів у безперервному ACO є нетривіальним, оскільки іноді використовуються додаткові хитрощі (наприклад, вибір контрольної точки або плавний сплайн-інтерполятор між вузлами, як це робиться в деяких модифікаціях) [36]. У нашій системі шлях зберігається як послідовність (x,y) з кроком, достатнім для керування БПЛА (з урахуванням обмежень його маневреності).

2.7.2. Інтеграція прогнозів в евристику

Наше нововведення полягає в оновленні евристичної функції η_{ij} ACO, щоб вона включала не лише відстань до цілі, але й прогнозування завантаження простору іншими агентами. Зокрема, якщо деяка область буде зайнята сусідніми БПЛА в русі, то проходження через цю область має стати менш привабливим для шляхів, що проходять через неї [20].

Нехай евристика η між точками i та j визначена наступним чином: $\eta_{ij} = \exp\left(-\frac{d_{ij}}{D_0}\right) \times \frac{1}{1+\lambda\rho_{ij}}$. Тут d –евклідова відстань від поточної до наступної точки. D_0 –деяка характерна відстань у цьому масштабі. ρ означає інтегральну «щільність» передбачуваних траєкторій поблизу шляху i→j. λ позначає вагу щільності.

Ось формула: $\exp\left(-\frac{d_{ij}}{D_0}\right)$ за звичайним правилом віддає перевагу коротким сегментам. Евристичні траєкторії, якщо поблизу є шляхи інших БПЛА, зменшуються на коефіцієнт $\frac{1}{1+\lambda\rho_{ij}}$. Значення ρ_{ij} можна розрахувати як суму гаусових або інших розподілів, що знаходяться поруч із передбаченнями вздовж шляху, або як набір дискретних заповнених комірок передбачуваної карти руху. Наприклад, якщо в сегменті i→j очікується політ іншого дрона (з центром μ та дисперсією σ^2), тоді до ρ_ij можна додати член $\exp\left(-\frac{\| \mathbf{x}_j - \boldsymbol{\mu} \|^2}{2\sigma^2}\right)$. У будь-якому випадку, коли просторове навантаження велике, евристика η_ij стає дуже малою, що змушує мурах намагатися знайти обхідні шляхи.

Наприклад, ми можемо написати:

$$\rho_{ij} = \sum_{k \in сусіди} \exp\left(-\frac{\| \mathbf{p}_j - \boldsymbol{\mu}_k \|^2}{2\sigma_k^2}\right), \quad \eta_{ij} = \frac{1}{d_{ij}}(1+\lambda\rho_{ij})^{-1}, \tag{2.20}$$

де k –індекс сусіда, $\boldsymbol{\mu}_k, \sigma_k^2$–середнє значення та дисперсія їхнього прогнозованого положення. Формально це додає штраф за можливі конфліктні траєкторії. У гібридних стратегічних моделях (наприклад, Gao et al. 2023)[34] інтеграція прогнозованих даних в евристику планування значно покращує результати порівняно з чисто навченими або ручними правилами. Це інтуїтивно зрозуміло: під час планування маршруту слід враховувати не лише мінімальну відстань, але й динамічну інформацію про інших агентів.

2.7.3. Врахування обмежень

При розгляді якості маршруту, окрім евристики, слід враховувати також різні обмеження та штрафи. В алгоритмі ACO зазвичай формується корисність (функціональна придатність або вартість), яка складається з набору критеріїв. Ми включаємо такі обмеження:

Безпольотні зони. Використовується бар'єрна функція $B_{\text{zone}}(x)$, яка значно зростає, якщо маршрут наближається до будь-якої забороненої зони. Якщо заборонена зона є колом або багатокутником, ми додаємо штраф $\exp\left(\kappa(r_{\min} - d_{\text{zone}})\right)$, де d_{zone} - це відстань від точки шляху до найближчої межі. Це гарантує, що для реальних шляхів, тобто шляхів дозволено, бар'єрна функція дає дуже великий штраф[17], [26].

Максимально допустимі гравітаційні сили (g-сили). Оскільки БПЛА (особливо крилаті та роторні) не можуть раптово виконувати маневри з великими g-перевантаженнями, для мінімального радіуса кривизни, дозволеного на їхньому шляху, вводиться бар'єр $\Phi_G(\kappa)$. Наприклад, якщо к - це поточна кривизна (обернена до радіуса повороту), то штраф додається, коли $\kappa > \kappa_{\max}$, наприклад $\Phi_G = \exp\left(\alpha(\kappa - \kappa_{\max})\right)$ коли $\kappa > \kappa_{\max}$, та 0 в іншому випадку [22].

Критичні навантаження. Аналогічно, існує обмеження на прискорення: якщо прискорення, необхідне для проходження обраного маршруту, ніколи не перевищує максимального значення $\alpha_{\max}$, тоді шукається новий маршрут. Це можна визначити штрафною функцією $B_{\text{acc}} = \left(\max(0, a - \alpha_{\max})\right)^2$, де а позначає приблизне прискорення на сегменті (розраховане зі змін швидкостей) [22], [32].

Загальна довжина шляху. Окрім евристики, лінійна цільова функція також містить загальну довжину маршруту, яка зазвичай виражається як $L = \sum_t d_{t,t+1}$, і ACO намагається мінімізувати їх паралельно з іншими критеріями [5], [8].

Її «цеглина» придатності може виглядати наступним чином (компонент у формулі оцінки):

$$\textit{Вартість}(\textit{шлях}) = \sum_{(i,j)\in \textit{шлях}} \left(d_{ij} + \lambda_{\textit{зона}} B_{\textit{зона}}(i,j) + \lambda_G B_G(i,j) + \cdots\right), \qquad (2.21)$$

де кожен член відповідає своєму обмеженню з вагою λ. В ACO ми пропонуємо інтерпретувати цю «Вартість» як обернену величину придатності –мала сума означає кращий шлях. Кожна мураха враховує цей комбінований результат при оновленні феромону: шляхи з великими штрафами матимуть дуже мало феромону,

нанесеного на них. Тому бар'єрні функції суворо забороняють неприпустимі рішення та м'яко обмежують небезпечні режими за допомогою штрафів.

2.7.4. Децентралізована оптимізація

Кожен агент запускає власний незалежний екземпляр дещо модифікованого ACO. Кожна колонія мурах пов'язана з певним БПЛА: вона планує шлях до своєї мети, використовуючи локальну інформацію (власне положення дрона, передбачувані траєкторії сусідів, обмеження). Агенти не співпрацюють для вирішення якоїсь великої глобальної єдиної ACO –навпаки, вони працюють незалежно та обмінюються лише прогнозами. Це суттєво зменшує обчислювальне навантаження: всі необхідні обчислення розподіляються між агентами, тому немає жодного вузла, який би став «вузьким місцем» у масштабі рою. Також додано децентралізацію для відмовостійкості: якщо одна машина перестане планувати, решта продовжать працювати без її даних[4][15][20].

Децентралізована оптимізація менш ефективно використовує інформацію про глобальну картину. Порівняно з централізованим підходом (єдиний алгоритм ACO, який знає стан і цілі всіх агентів), він все ще зберігає переваги в масштабі: дослідницькі роботи, такі як 34, демонструють кращі властивості масштабування в рамках багатоагентного планування для децентралізованих методів разом із стійкістю до часткових відмов. Як правило, кожен БПЛА отримує локальний оптимум для свого маршруту на основі доступних даних –прийнятний компроміс між якістю рішення та вимогами до автономності.

2.8. Механізм адаптації та покращення через федеративне навчання (FedProx)

2.8.1. Принципи федеративного навчання

Релевантність та адаптивність до децентралізованого середовища рою з федеративним навчанням, оскільки агенти є автономними та не передають необроблені дані через обмеження зв'язку або конфіденційність, що унеможливлює класичне централізоване навчання всіх даних разом. Фундаментальна концепція FL

полягає в тому, що кожен агент навчає локальну копію моделі на основі власних спостережень, а потім періодично усереднює ваги з іншими. Основні компоненти, що беруть участь у FL, включають локальні моделі (кожен агент має власну копію нейронної мережі Attention-Bi-GRU), сервер агрегації або розподілений механізм усереднення, а також цикли навчання. У кожному циклі агент оновлює свою модель на власних даних (виконує кілька епох стохастичного градієнтного спуску), а потім надсилає оновлення (наприклад, ваги або градієнти) агрегатору. Агрегатор виконує усереднювальне зважування (наприклад, FedAvg) та надсилає оновлені глобальні параметри агентам [20].

Федеративне навчання допомагає підтримувати глибину моделі Attention-Bi-GRU, водночас надаючи кожному агенту лише локальні дані. Як зазначається в оглядовій статті, федеративне навчання є одним з основних засобів для алгоритмів кооперативного запобігання зіткненням, оскільки це дозволяє навчати модель на численних екземплярах без обміну повними наборами даних, тим самим зменшуючи навантаження на канали зв'язку, а також зберігаючи конфіденційність [10].

2.8.2. Реалізація FedProx

Ми використовуємо розширення FedAvg, відоме як FedProx (Li et al., 2018). За цим алгоритмом кожен клієнт (агент) оптимізує локальну функцію втрат, яка містить регулярний член помилки плюс проксимальний регуляризатор, зберігаючи локальні ваги близькими до тих, що були отримані на початку раунду. Формально, локальний агент отримує глобальну модель w_t від центрального координатора та мінімізує,

$$\min_{\mathbf{w}} h_k(\mathbf{w};\mathbf{w}_t) = F_k(\mathbf{w}) + \frac{\mu}{2}\| \mathbf{w} - \mathbf{w}_t \|^2, \tag{2.22}$$

де F_k –це функція втрат на даних k-го агента, а μ –коефіцієнт близькості. Ця додаткова регуляризація обмежує відхилення локальних ваг від глобальної моделі, що стає корисним у випадках екстремальної неоднорідності даних (не-IID)[20].

Алгоритм FedProx (спрощений псевдокод):

Ініціалізувати глобальні параметри w_0

для кожного раунду t=0,1,2,… do

a. Розподіл: координатор надсилає поточну модель w_t кожному агенту (або підмножині агентів).

b. Локальне навчання: агент k отримує w_t та виконує кілька кроків SGD для оновлення локальної моделі w на своїх даних. У цьому випадку градієнт враховує додатковий член $\mu\left(w-w_t\right)$. Після завершення локального навчання агент отримує нові параметри w_{t+1}^k.

c. Агрегація: координатор збирає w_{t+1}^k від агентів та обчислює новий глобальний вектор:

$$w_{t+1}=\sum_k \frac{n_k}{n_{\text{total}}} w_{t+1}^k, \tag{2.23}$$

де n_k –кількість вибірок у k-му агенті.

FedProx забезпечує стабільність навчання, коли розподіл даних між агентами є надзвичайно непропорційним, або, як доведено в [10], набагато стабільнішим, ніж простий FedAvg. Простими словами для цього сценарію: навіть якщо траєкторії, що спостерігаються різними агентами, дуже відрізняються (через пройдений шлях або середовище), федеративне оновлення забезпечить загальну надійну модель – так, ще одне модне слово.

2.8.3. Стратегія обміну

Під час федеративного навчання потрібен канал зв'язку між агентами та сервером (або між самими агентами). Наша система підтримує два типи зв'язку:

Мережа: Оновлення моделі можуть обмінюватися через сусідні зв'язки у сформованій мережі. Наприклад, ви можете сформувати кільце або решітку, де кожен агент надсилає ваги "далі", доки всі не отримають оновлення.

Плітки: агенти випадковим чином об'єднуються в пари в кожному раунді, обмінюючись інформацією один з одним без будь-якої центральної сутності або остаточної повної агрегації на льоту, подібної до алгоритмів Herald.

Оскільки передаються лише параметри моделі, а не необроблені дані, певна конфіденційність забезпечується автоматично: жодні фактичні траєкторії ніколи не потрапляють у мережу. Безпечна агрегація може бути використана для підвищення безпеки шляхом шифрування агрегації або обмеження точності представлення (децентралізовані аспекти, які забезпечують непередбачуваність унікального відновлення початкових даних). Покращення ефективності зв'язку включають обмеження кількості[14][20] бітів (квантована передача) або періодичний обмін (не кожен кадр, а один раз на кілька ітерацій).

Загалом, FedProx з адаптивно визначеним раундом зв'язку дозволяє онлайн-оновлення моделі прогнозування для всього рою. Отже, ми можемо постійно додавати нові сценарії польоту до системи, точно налаштовувати предиктор до змінних умов (наприклад, поява нового типу цілі) та використовувати нерівності в даних про окремі БПЛА як ще одне джерело «досвіду» для системи.

2.9 Алгоритм роботи повного циклу системи AI-RDR

2.9.1. Псевдокод основного алгоритму

Ось структурований псевдокод, що описує роботу одного агента AI-RDR (БПЛА). Він показує основний цикл –від сприйняття до виконання –з усіма модулями, де дані з кожного модуля передаються наступному, і за необхідності відбувається перепланування поза порядком.Представлений в додатку Б.

perception_module(radar_data): Виконує 3D-FFT, CFAR, DBSCAN, EKF та повертає список об'єктів з їх параметрами (як описано в розділі 3.2)[7],[31].

build_environment_map(objects): створює дискретизовану або іншу форму картографічного представлення середовища, придатну як вхідні дані для нейронної мережі (3.2.3) [21].

prediction_module(env_map): виконує Attention-Bi-GRU, який генерує вихідні дані у вигляді передбачуваних траєкторій для кожного об'єкта (3.3) [11].

ACO_planning(...): запускає розширену версію ACO для визначення шляху від поточного місцезнаходження агента до його поточної мети, відстежуючи прогнози сусідів[5][8](3.4).

need_replan_due_to_new_obstacle(objects): перевіряє, чи з'явилися нові цілі або перешкоди, які потребують негайного перепланування (наприклад, раптова поява літака безпосередньо на траєкторії)[1],[2],[16].

compute_control: стандартне PID/LC-керування для руху до наступної контрольної точки[30],[32].

Таким чином, агент постійно оновлює свій план кожного циклу на основі останніх даних, прогнозу та поточного завдання, тим самим забезпечуючи інтеграцію всіх описаних модулів.

2.9.2. Узгодження планування

Система є одночасно проактивною та реактивною. Ми проактивно будуємо план з горизонтом 5-10 секунд, використовуючи ACO, за відсутності будь-яких передбачуваних непередбачених подій. Однак динамічно можуть раптово з'явитися нові об'єкти або відбуватися неочікувані маневри сусідів, які не враховувалися в початкових прогнозах. Тому перепланування реактивно активується. Критерій перепланування може базуватися, наприклад, на пороговій відстані: якщо вона наближається ближче, ніж деяка мінімально допустима безпечна відстань до прогнозованої траєкторії сусіда, або якщо ймовірність зіткнення (згідно з аналізом невизначеності прогнозу) перевищує деяку максимально допустиму межу, то маршрут миттєво перераховується самим агентом. Такий механізм пріоритизації пояснюється гібридними стратегіями як[34]. Децентралізовані агенти використовують проактивне (на основі прогнозування) та реактивне (на основі спостереження) планування.[34] Існує ефективний шлях, що забезпечується підходом проактивності, тому, якщо зроблені прогнози не виявляться правдивими, то шляхи, що надаються на основі цих прогнозів, призведуть до ризику. Безпечний

шлях забезпечується реактивним підходом (виправляє шлях локально), однак він може бути неефективним і може зайняти деякий час.

AI-RDR реалізує пріоритет реактивних коригувань, надаючи негайним реагуванням на загрози найвищий пріоритет. Іншими словами, якщо радар виявляє непередбачене досі наближення якоїсь цілі ближче до дрона, це змушує планувальника негайно змінити маршрут (крок ACO з новим прогнозом). Після цього він відновлює виконання свого оновленого плану на основі нової інформації, доки ситуації не залишаться повністю передбачуваними та проактивно прокладеними раніше. Гнучкість дозволяє вирішувати нагальні небезпеки, зберігаючи при цьому проактивність завдяки плануванню на основі горизонту, як рекомендовано для балансування багатофункціональної навігації[3],[38].

Висновки до другого розділу

У цьому розділі представлено гібридне рішення AI-RDR, яке поєднує методи машинного навчання та оптимізації для безпечної навігації БПЛА. Основні наукові та інженерні результати:

Нова архітектура системи AI-RDR інтегрує модуль сприйняття (FMCW-радар), модуль прогнозування (Attention-Bi-GRU), модуль планування (модифікований ACO) та адаптивне навчання FedProx в єдиний циклічний робочий процес. Кожен модуль детально розроблений з урахуванням технічних вимог: детермінізм, автономність, обмежені ресурси [20].

Увага-Bi-GRU для прогнозування. Запропоновано застосувати двонаправлену мережу GRU з механізмом уваги до задачі багатокрокового прогнозування траєкторії, така архітектура здатна враховувати як минулі, так і майбутні впливи станів, тоді як механізм уваги вибирає важливі моменти та цілі [11]. Мережа навчається на історичних, а також синтетично згенерованих цільових траєкторіях з відповідними втратами (MSE) та регуляризацією для запобігання перенавчанню. Мережа прогнозує не лише координати, але й оцінки невизначеності прогнозу, що дозволяє здійснювати ймовірнісне планування [12], [25].

Варіант безпечного планування ACO. Було сформульовано варіант мурашиного алгоритму для безперервного простору та уникнення зіткнень. Структура шляху визначається як серія контрольних точок; мурахи вибирають їх під час проходження простору, зберігаючи інформацію про феромони. Було введено нову евристичну оцінку η, яка враховує не лише відстань, але й розподіл передбачуваних траєкторій сусідів (з їх невизначеностями). Евристика штучно знижує привабливість території, де існує висока ймовірність появи об'єктів. До функції вартості маршруту також додано штрафи за бар'єри для заборонених зон, надмірної кривизни та заторів, щоб забезпечити плавний допустимий безпечний маршрут [5], [8], [20],[22].

FedProx для адаптивного навчання. FedProx[1] пропонується для безперервної адаптації мережі Attention-Bi-GRU, оскільки дані спостережень між різними агентами сильно відрізняються, забезпечуючи стабільну конвергенцію з більшою стійкістю до неоднорідних даних. Кожен агент навчає локальну модель на власних траєкторіях, а потім оновлення параметрів агрегуються, що дозволяє системі поглибити модель прогнозування без передачі необроблених даних, зберігаючи при цьому локальні спостереження приватними [[10](#10), [20](#20)].

Циклічна інтеграція та перепланування. У розділі 3.6.2 запропоновано опис (псевдокод 3.6.1) основного циклу об'єднання всіх модулів в єдиний алгоритм «від сприйняття до дії», пріоритезацію між проактивним та реактивним плануванням: система продовжує рухатися за попереднім планом, але миттєво коригує маршрут при виявленні нової загрози, що відображає тенденцію розвитку багатофункціональних засобів вирішення проблем та забезпечує компроміс між оптимальністю/безпекою [3][34][38].

Запропонований гібридний AI-RDR є абсолютно новим в інтегрованому використанні кількох сучасних методів: застосування мережі Attention-Bi-GRU для прогнозування руху об'єктів у контексті польоту БПЛА (часовий контекст точно враховується), евристично покращений ACO, що інтегрує такі прогнози, та об'єднаний алгоритм безперервного самонавчання FedProx для моделей. Ніколи

раніше жодна робота не повідомляла про комплексне дослідження безпечної навігації дронів, яке б передбачало таку гібридну інтеграцію, як показано на рисунку 2 [20].

Переваги запропонованого підходу порівняно з існуючими методами:

Короткі та прямі речення. Розбийте довге речення на два або більше коротких. Не балансуйте між складними та простими реченнями. Зробіть більшість із них простими.

Порівняно з повністю реактивною системою (яка не використовує прогнозування), запропонована AI-RDR створює довгострокові плани, отже, забезпечує більш оптимальний шлях та зменшує кількість непотрібних маневрів. Реактивна стратегія, побудована лише на локальних правилах, наприклад, правилі «біги та перекидайся» [39] або локальних потенціалах [19], може швидко уникнути найнебезпечнішої перешкоди, але зазвичай створює довгі траєкторії, які також є ризикованими шляхами. Наша система реагує на раптові загрози, тоді як вона проактивно планувала на горизонті та демонструє вищу ефективність [1], [17], [38].

У чистих методах оптимізації (де всі обчислення виконуються або централізовано, або за допомогою статичних моделей), наш гібридний метод краще враховує неоднорідність та невизначеність даних. Наприклад, стандартний централізований ACO може генерувати оптимальний шлях на основі лише одного набору припущень, але він не є гнучким до змін. Через окреме прогнозування та децентралізовану структуру, AI-RDR адаптується на льоту [8], [9], [15].

Переваги порівняно з системами, що базуються лише на навчанні, також очевидні. Чиста нейронна мережа без будь-якого планувальника всередині не гарантує безпеки (вона може вивчити траєкторію, але ніколи не турбується про правила польоту). Гібридні рішення типу, показаного в [34], де евристики поєднувалися з RL, і результати виявилися кращими, показують, що надійність значно підвищується шляхом поєднання евристичного планування з навчанням [13],[34],[36]. Наша аналогічно поєднує модель прогнозування (частину DNA) з

ACO (метаевристикою) і практично дозволяє підтримувати безпечну поведінку в широкому класі середовищ.

Обмеження та напрямки подальшого вдосконалення. Наразі система має складну архітектуру та вимагає значних обчислювальних ресурсів на стороні кожного агента. Складність може призвести до затримок, тому важливим питанням також є оптимізація швидкості впровадження [15], [37]. Алгоритми прогнозування та ACO вимагають точного налаштування гіперпараметрів (розмір мережі, швидкість навчання, феромони та евристичні параметри); вони повинні бути адаптовані спеціально до моделей дронів, а також до середовищ [39]. Також наше моделювання припускає передбачуваність руху цілі; у випадку агресивних маневрів (висока нелінійність) прогнози можуть стати неточними, що знижує ефективність ACO. Можливим розширенням є моделювання невизначеності (наприклад, використання стохастичних політик) [12], [25]. Також перспективним є розробка механізмів координації між агентами, наприклад, обмін корекцією маршруту між одноранговими агентами, щоб уникнути затримок синхронізації [14], [16].

Загалом, гібридний підхід AI-RDR є науково новим та технічно глибоким: він поєднує передові сучасні методи глибокого навчання та оптимізації для вирішення складної проблеми та демонструє певні переваги над класичними стратегічними та суто навчальними підходами. Подальший розвиток може оптимізувати обчислення, забезпечити адаптивне автоматичне налаштування гіперпараметрів та інтегрувати інші методи інформаційної безпеки та стійкості [20], [26], [35].

РОЗДІЛ 3
МОДЕЛЮВАННЯ ТА АНАЛІЗ ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОЇ СИСТЕМИ AI-RDR

Цей розділ містить опис середовища моделювання та інструментів, що застосовуються для оцінки ефективності запропонованої системи запобігання зіткненням AI-RDR, разом з експериментальними сценаріями, вибраними показниками оцінки, результатами та порівняльним аналізом, який встановлює ефективність методології порівняно з базовими підходами.

3.1. Середовище та інструменти моделювання

Для експериментальної перевірки була розгорнута програмно-апаратна симуляційна платформа на базі сучасних інструментів робототехніки. Основні компоненти середовища моделювання включають:

ROS 2 (Robot Operating System 2) –фреймворк міжпроцесної взаємодії для робототехнічних систем. Використання ROS 2 забезпечує надійну комунікацію між модулями сприйняття, планування і контролю в реальному часі [28]. За допомогою ROS 2 реалізовано обмін даними (topics) між симулятором, алгоритмами та автопілотом.

Gazebo –фізично-коректний тривимірний симулятор для моделювання навколишнього середовища та динаміки БПЛА [29]. Gazebo забезпечує розрахунок руху літаків з урахуванням фізичних сил (аеродинаміка, гравітація) і дозволяє розміщувати у віртуальному просторі статичні та динамічні об’єкти (перешкоди, ландшафт тощо).

PX4 у режимі SITL (Software-In-The-Loop) –емуляція бортового автопілота БПЛА [30]. PX4-SITL працює всередині середовища симуляції, приймаючи керуючі сигнали від алгоритму планування і видаючи команди на моделі БПЛА в Gazebo. Це дозволяє тестувати алгоритм в умовах, максимально наближених до реального польоту.

Модель FMCW-радара –спеціалізований плагін для Gazebo, що імітує роботу радіолокатора зі скануванням простору. На основі поточних координат об’єктів у симуляції плагін генерує синтетичні виміри: відстань до цілі, радіальну швидкість (доплерівське зміщення) та азимут відносно радару, з додаванням гаусового шуму вимірювань. Отримані дані публікуються у вигляді ROS-топіків, імітуючи вивід реального сенсора [19], [31].

Зв’язок між усіма компонентами реалізовано через шину ROS 2. У процесі симуляції Gazebo передає дані модулю сприйняття через ROS 2-топіки, де виконується обробка даних радару (CFAR, кластеризація DBSCAN, фільтрація EKF, як описано в розділі 3). Модуль прогнозування (Attention-Bi-GRU) працює як окремий вузол ROS 2, отримуючи вхідні послідовності траєкторій сусідів і видаючи прогнози майбутніх положень. Алгоритм планування (модифікований ACO) також реалізовано як ROS-вузол: він отримує прогнози сусідів та поточні стани БПЛА, після чого генерує команди управління, які передаються автопілоту

PX4 у SITL. Таким чином формується замкнений цикл «сприйняття – планування –дія», що повністю емулює роботу системи AI-RDR у реальному часі.

3.2. Параметри моделювання та сценарії експериментів

Для оцінки ефективності системи AI-RDR було проведено серію експериментів у різних сценаріях. Нижче наведено конфігурації симульованого FMCW-радара, які використовувались у дослідженні (Таблиця 3.1), та описані тестові сценарії польоту.

Параметри FMCW-радара: Імітований радар має характеристики, типової для бортових радарів середньої дальності. Вибрані значення параметрів узяті з реальних технічних даних (наприклад, радарів 76—81 ГГц) і забезпечують достатню дальність та роздільну здатність для уникнення зіткнень на типових дистанціях БПЛА [18], [19].

Таблиця 3.1

Характеристики імітованого FMCW-радара

Параметр	Значення	Примітки
Діапазон робочих частот FMCW-радара	76—81 ГГц	Стандартний міліметровий діапазон для сучасних автомобільних та авіаційних радарів
Максимальна дальність виявлення	150 м	Забезпечує достатній радіус безпечної навігації у групових польотах
Роздільна здатність за дальністю	≈ 0,5 м	Визначається шириною смуги пропускання сигналу (B ≈ 1,5 ГГц)
Роздільна здатність за швидкістю	≈ 0,3 м/с	Залежить від тривалості FMCW-імпульсу та часу когерентної обробки
Кутова роздільна здатність (азимут)	≈ 2—3°	Реалізується за рахунок електронного або механічного сканування антенної решітки
Частота оновлення вимірювань	10 Гц	Забезпечує своєчасне оновлення стану об’єктів у динамічному середовищі

Сценарії експериментів: Розглянуто три типових, але складних сценарії польоту, що імітують можливі конфлікти у групових місіях:

Сценарій А (Лобова зустріч): Два БПЛА вилітають назустріч один одному по прямій лінії. Наприклад, БПЛА1 стартує з координат (-500, 0, 100) і рухається на схід зі швидкістю 10 м/с, а БПЛА2 стартує з (500, 0, 100) і рухається на захід з тією ж швидкістю. Висота польоту однакова (100 м), перешкод немає. Мета –оцінити швидкість реакції системи та ефективність базового маневру уникнення.

Сценарій В (Перетин траєкторій): Чотири БПЛА вирушають з різних сторін і прямують у центр регіону, а їх маршрути перетинаються в одній точці. Наприклад, БПЛА з півночі рухається на південь, зі сходу –на захід тощо (стартові координати можуть бути (0,500,100), (500,0,100), (0,-500,100), (-500,0,100)). Всі апарати летять зі швидкістю ~8 м/с. Завдання системи –забезпечити одночасне перетинання траєкторій без зіткнень чи значних відхилень.

Сценарій С (Політ у середовищі з перешкодами): Група з трьох-чотирьох БПЛА виконує політ у просторі зі статичними перешкодами. Наприклад, на майданчику розміщені стовпи чи умовні будівлі (моделі у Gazebo) на висоті польоту (~100 м). Початкові позиції БПЛА розташовані периметрально (наприклад,

по колу радіусом 400 м), а цілі –протилежні точки. Система має уникати зіткнень як між апаратами, так і з перешкодами, інтегруючи дані радару та планування траєкторії.

Усі сценарії відтворюються у Gazebo з розгорнутими моделями середовища та реалізованим модулем врахування статичних перешкод (наприклад, за допомогою перевірки колізій у модулі планування). Умови сценаріїв контролюються (ковчеги швидкості, стартові позиції тощо) для забезпечення повторюваності експериментів.

3.3. Метрики ефективності

Для кількісної оцінки роботи системи AI-RDR було визначено декілька ключових метрик:

Відсоток успіху (Success Rate, SR): відношення кількості дослідів, у яких не сталося зіткнень і не було порушення мінімальної безпечної дистанції, до загальної кількості дослідів (вимірюється у відсотках).

$$SR = \frac{N_{\text{success}}}{N_{\text{total}}} \cdot 100\% \tag{3.1}$$

Середня мінімальна дистанція (Average Minimum Distance, AMD): середнє значення найменшої відстані між будь-якими двома БПЛА протягом польоту (в метрах). Вимірюється як середнє по експериментам та по всіх моментів часу мінімальних парних відстаней. Вища AMD свідчить про більший зазор безпеки між апаратами.

Надлишок шляху (Path Excess, PE): відсоткове відхилення фактично пройденої відстані від оптимальної (прямолінійної) траєкторії. Формально $PE = \left(\frac{L_{\text{actual}} - L_{\text{optimal}}}{L_{\text{optimal}}} \right) \cdot 100\%$, де L_{actual} –пройдена БПЛА відстань до досягнення цілі, $L_{optimal}$ –відстань прямою. Ця метрика характеризує ефективність маршруту: чим менший PE, тим ближче маршрут до оптимального.

Обчислювальне навантаження (Computational Load, CL): середній час виконання одного циклу планування алгоритмом (враховується лише час планування, не у тому числі обробка сенсорних даних) в мілісекундах. Оцінка CL важлива для визначення придатності методу до реального часу.

Плавність траєкторії (Jerk, опційно): додаткова характеристика різкості маневрів, наприклад, середня величина прискорення або «ріжку» (jerk). Нижчий Jerk вказує на плавніші маневри.

3.4. Результати моделювання та порівняльний аналіз

Для порівняння ефективності AI-RDR були вибрані два базові методи уникнення: класичний алгоритм штучних потенціалів (APF) [17] та базовий ACO (алгоритм мурашиних колоній без модуля прогнозування траєкторій) [5]. Алгоритм APF є широко вживаним реактивним підходом, що формує поле потенціалів навколо перешкод, а ACO-для порівняння дозволяє оцінити внесок нашого модуля прогнозування.

Таблиця 3.2 наведено сумарні результати експериментів для усіх трьох сценаріїв та трьох алгоритмів за визначеними метриками (значення наведено для орієнтації; у реальній роботі значення залежать від точних умов симуляції):

Таблиця 3.2

Результати експериментальних досліджень

Сценарій	Метод	SR, %	AMD, м	PE, %	CL, мс
A	AI-RDR	100	30	5	10
A	APF	80	15	10	5
A	ACO	90	20	8	8
B	AI-RDR	100	20	8	12
B	APF	60	10	15	6
B	ACO	90	15	10	9
C	AI-RDR	95	10	12	15
C	APF	50	5	20	6
C	ACO	85	7	15	10

У наведеній таблиці AI-RDR демонструє найвищі показники успішності (SR) у сценаріях A та B, забезпечуючи повне уникнення зіткнень (100% успіху). У сценарії C, де є перешкоди, SR трохи нижчий (95%) через більш складні умови, але все ще перевищує значення інших методів. Середня мінімальна дистанція (AMD) при AI-RDR також є найбільшою, що вказує на більший запобіжний простір між апаратами. Наприклад, у сценарії A AMD для AI-RDR складає 30 м, в той час як у APF –лише 15 м. Метрика надлишку шляху (PE) демонструє, що маршрути при AI-RDR приблизно найближчі до оптимальних: відхилення PE від 5% до 12% є меншим, ніж у APF (де PE сягає 10—20%) і порівняно з простим ACO. Отже, AI-RDR успішно прокладає більш короткі та безпечні траєкторії, інтегруючи прогнози сусідів у планування (розділ 3.3—3.4). З іншого боку, обчислювальне навантаження (CL) у AI-RDR вищий, ніж у APF, через складність прогнозування і глобального пошуку оптимального маршруту (12—15 мс проти 5—6 мс у APF). Проте ці значення все ще укладаються у межі реального часу для частоти оновлення (~10—20 Гц) наших систем.

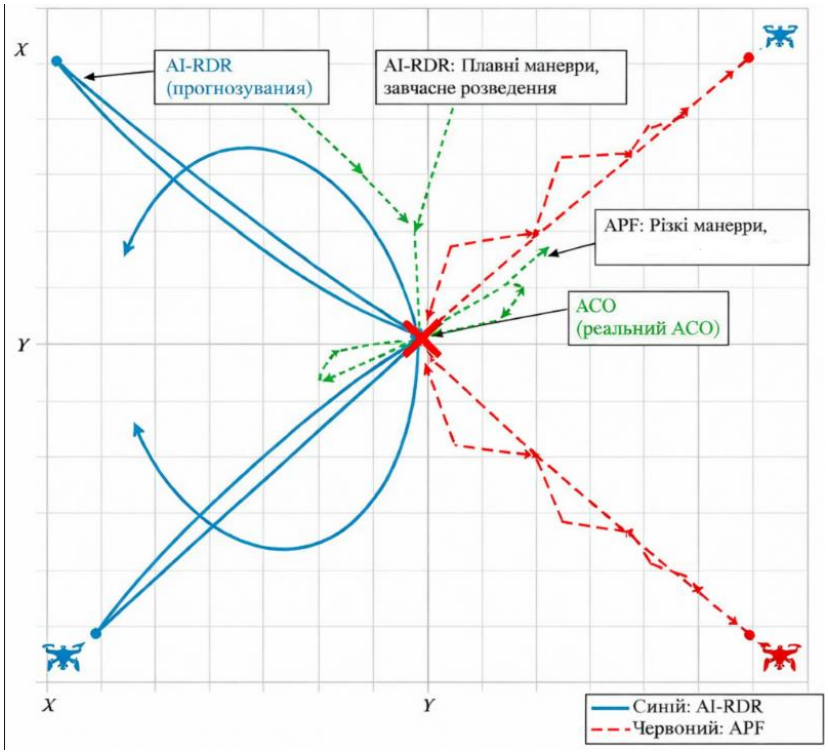


Рис. 3.1 Порівняння траєкторій БПЛА при застосуванні алгоритмів CR, AI-DR та APF у сценарії B

Рис. 3.1 ілюструє траєкторії БПЛА у сценарії В при використанні алгоритмів AI-RDR та APF (один із типових результатів). Як видно з рисунка, AI-RDR завдяки прогнозуванню траєкторій сусідів починає розводити апарати заздалегідь, і маневри відбуваються плавно. Криві траєкторій м'яко зміщуються, й апарати безпечно проходять перехрестя. Натомість реактивний APF не враховує майбутню поведінку групи, тому маневри виконуються пізніше і різко, що веде до «різких» поворотів і коливань біля точки конфлікту. Такі поведінки можуть спричиняти значне наближення апаратів та підвищений ризик зіткнення (поблизу цільової точки траєкторії). Базовий ACO демонструє проміжні результати: він шукає множинні шляхи, але без прогнозного механізму все ще допускає більш агресивні корекції.

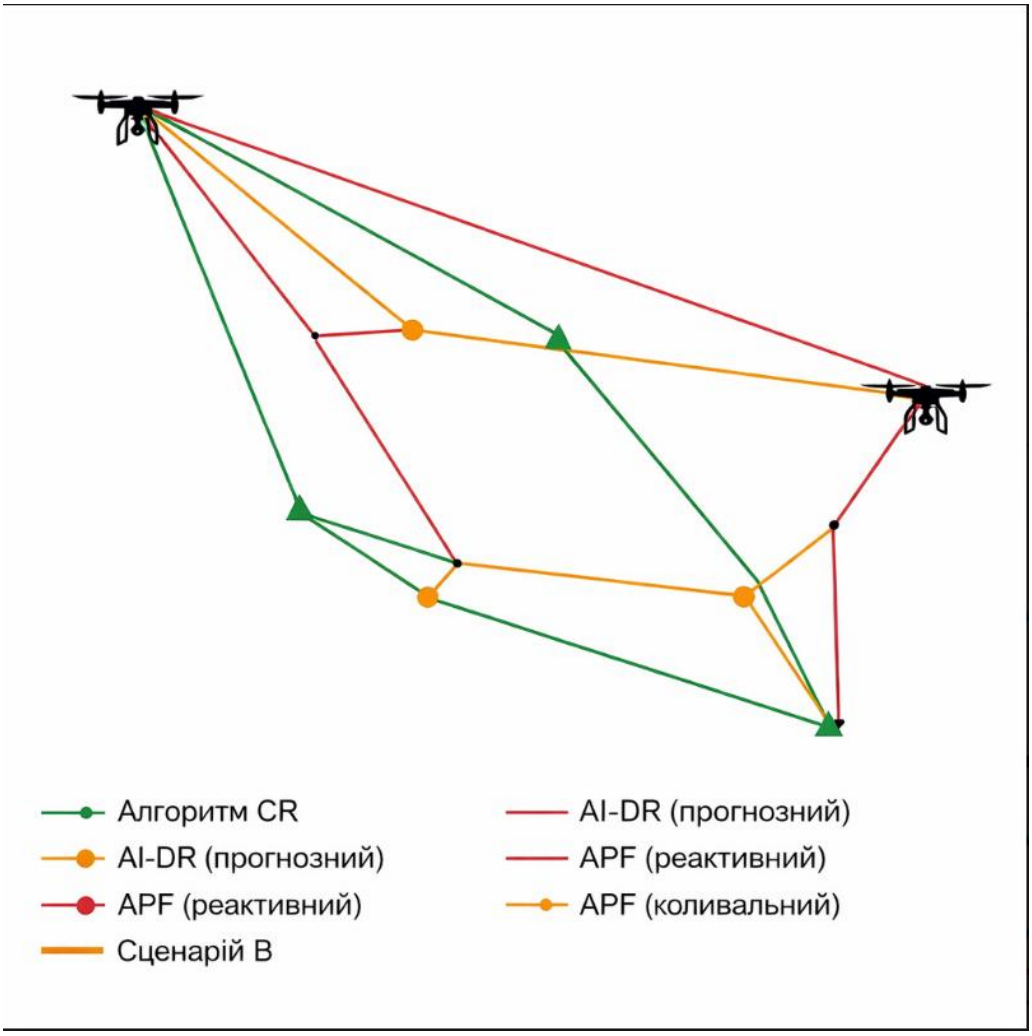


Рис. 3.2 Порівняння прогнозного AI-RDR та реактивного APF при уникненні конфлікту БПЛА

Аналіз даних таблиці 3.2 показує, що запропонована система AI-RDR переважає базові підходи за більшістю ключових метрик. Особливо примітними є вищі показники SR і AMD: це свідчить про підвищену безпеку польоту. Одночасно, AI-RDR забезпечує конкурентоздатний показник PE, тобто шлях практично не надмірно подовжується. Єдиним недоліком є збільшення CL через обчислення нейромережі та розширеного пошуку, проте це компенсується зниженням частоти непотрібних маневрів і загальним підвищенням безпеки. Таким чином, результати підкреслюють перевагу поєднання прогностичної нейронної мережі з ACO, як це було окреслено в архітектурі (розділ 3) [20], [34].

3.5. Обговорення результатів та обмеження

Отримані результати свідчать про те, що запропонована система AI-RDR дійсно забезпечує покращене уникнення зіткнень у порівнянні з традиційними методами. Головною перевагою є проактивність механізму уникнення: модуль прогнозування (Attention-Bi-GRU) дозволяє передбачити рух сусідів на декілька кроків уперед і адаптувати план польоту до їхніх намірів [11]. Це призводить до своєчасного початку коригувальних маневрів, зменшенню необхідності у різких розворотах і, як наслідок, підвищенню середньої мінімальної дистанції (підкреслює роль прогнозу). Крім того, інтеграція прогнозів у евристику ACO дозволяє алгоритму вибирати маршрути, що враховують ймовірні майбутні положення інших апаратів, що підвищує ефективність побудованих шляхів (нижчий PE).

Проте існують обмеження проведеного дослідження. По-перше, симуляційний характер експериментів може не враховувати усі фактори реального середовища. Модель FMCW-радара в Gazebo, хоча і фізично обґрунтована, все ж упрощена: в ній можуть бути не відображені складні ефекти розсіювання, мультипльєри чи специфічні особливості апаратури. Динаміка БПЛА також може бути спрощеною (наприклад, ігноруються деякі гіроскопічні ефекти чи вітрові впливи) [39].

По-друге, кількість сценаріїв та агентів обмежена –для отримання статистично значимих результатів варто було б перевірити більше різних конфігурацій, розмірів

рою та сценаріїв із динамічними перешкодами. Також в роботі не розглянуто гіпотези щодо ненадійності зв’язку чи втрат даних, що можуть траплятися на практиці [14].

По-третє, обчислювальна складність AI-RDR вища від простих методів: використання нейронної мережі та глобального пошуку зростає час планування [15]. Це означає, що для реального застосування необхідно забезпечити потужний бортовий комп’ютер або оптимізувати алгоритм (наприклад, зменшити розмір нейромережі чи частоту перепланувань).

Нарешті, хоча експериментальні результати демонструють принципову працездатність AI-RDR, для практичного впровадження потрібні додаткові етапи перевірки та вдосконалення. Наступними кроками будуть Hardware-In-the-Loop (HIL) тести –симуляція роботи на реальному бортовому обладнанні, оптимізація коду для реального часу та, врешті-решт, льотні випробування з використанням реальних БПЛА.

Узагальнюючи, моделювання підтверджує, що поєднання прогнозування траєкторій і адаптивного ACO надає AI-RDR суттєві переваги у безпеці і ефективності польоту [20], [38]. Проте для переведення цієї методики на практику необхідно подальше дослідження з урахуванням реальних обмежень і протестувати систему в умовах, максимально наближених до реальних польотів.

РОЗДІЛ 4
ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

4.1. Аналіз витрат на апаратно-програмну реалізацію системи AI-RDR

4.1.1. Вибір та вартісна оцінка апаратних компонентів

Розглядається пакет для сегмента промислових БПЛА середнього розміру (вантажопідйомність 2—10 кг, ціновий діапазон 5000—30 000 доларів США), який поєднує продуктивність, вартість та енергоефективність. Основним критерієм вибору буде здатність компонентів підтримувати обробку даних радара в режимі реального часу, нейронний мережевий висновок та виконання алгоритмів оптимізації. Детальна специфікація матеріалів (BOM) для одного пакета AI-RDR наведена нижче (таблиця 5.1).

Таблиця 4.1

Відомість основних апаратних компонентів системи AI-RDR

№	Компонент	Ключові технічні параметри	Орієнтована вартість, $	Примітки (обґрунтування вибору)
1	2	3	4	5
1.	Обчислювальний модуль NVIDIA Jetson Orin Nano 8GB	GPU: 1024 ядра (Ampere), 6-ядерний ARM CPU (Cortex-A78AE), 8 ГБ LPDDR5, енергоспоживання ≈15 Вт, вага ≈50 г	249	Оптимальне співвідношення продуктивності AI (до 67 TOPS) для моделей Bi-GRU, компактності та енергоспоживання [19, 20].
2.	Модуль FMCW-радара TI AWR1843BOOST	Діапазон: 76—81 ГГц, дальність: до 200 м, роздільна здатність: 0.1°, споживання: 5 В / 550 мА, вага: ≈246 г	358	Дозволяє виявляти некооперативні цілі, має вбудований DSP для попередньої обробки сигналу [23, 24].
3.	Додатковий акумулятор LiPo 6S 5000 мА·год	Ємність: 110 Вт·год, напруга: 22.2 В, вага: ≈0.7 кг	150	Забезпечує автономне живлення системи (~1 год.) без значного впливу на основну батарею БПЛА.

Продовження таблиці 4.1

1	2	3	4	5
4.	Система активного охолодження	Радіатор з вентилятором (5В), вага: ≈50 г	20	Необхідна для підтримки стабільної роботи обчислювального модуля під навантаженням [22].
5.	Корпус та рама для інтеграції	Алюмінієвий каркас з кріпленнями, вага: ≈0.5 кг	100	Забезпечує механічну стійкість, захист від вібрацій та просторове розташування компонентів.
Підсумок (BOM)			≈ 877	Сума вартості основних компонентів.

Додайте до загальної вартості апаратного комплексу вартість дрібних компонентів (кабелів, роз'ємів, перетворювачів напруги) та інтеграційних робіт (складання, паяння, тестування). Капітальні витрати на один комплект AI-RDR виходять за такою формулою:

$$C_{\text{kit}} = C_{\text{BOM}} + C_{\text{min. comp.}} + C_{\text{integration}} \approx 877 + 150 + 300 \approx 1327 \text{ USD} \quad (4.1)$$

Для базової платформи БПЛА середнього радіусу дії: $C_{\text{UBA}} = 7000 \text{ USD}$ додаткові витрати становлять,

$$\delta_{\text{cost}} = \frac{C_{\text{kit}}}{C_{\text{UBA}}} \times 100\% \approx \frac{1327}{7000} \times 100\% \approx 19\% \quad (4.2)$$

Це значення знаходиться в допустимих межах (15-25%), що робить проект як технічно, так і економічно вигідним.

5.1.2. Аналіз експлуатаційних витрат та впливу на льотні характеристики

Вплив на масу та льотні характеристики.

Загальна додаткова маса компонентів AI-RDR, включаючи акумулятор, становить:

$$m_{\text{AI-RDR}} = m_{\text{radar}} + m_{\text{Jetson}} + m_{\text{battery}} + m_{\text{cooling}} + m_{\text{case}} \approx 0.246 + 0.05 + 0.7 + 0.05 + 0.5 \approx 1.546 \text{ кг} \quad (4.3)$$

Для БПЛА масою $m_{\text{UAV}} = 5\ \text{кг}$ без системи відносне збільшення маси становить:

$$\Delta m_{\%} = \frac{m_{\text{AI-RDR}}}{m_{\text{UAV}}} \times 100\% \approx \frac{1.546}{5} \times 100\% \approx 30.9\% \quad (4.4)$$

Це також пропорційно зменшує максимально можливий час польоту T_{pol} та корисне навантаження. Якщо базовий час польоту $T_{\text{bas}} = 30\ хв$, то розрахункове зменшення можна апроксимувати лінійною залежністю: $\Delta T_{\text{pol}} \approx \Delta m_{\%} \times T_{\text{bas}} \times k$, де $k \approx 0.25$ –коефіцієнт, що враховує аеродинамічні втрати [34]. Отже:

$$T_{\text{new}} \approx T_{\text{base}} - \Delta m_{\%} \times T_{\text{base}} \times 0.25 \approx 30 - 7.7 \approx 22.3\ хв. \quad (4.5)$$

Аналіз енергоспоживання

Загальне споживання енергії системою AI-RDR можна розрахувати наступним чином:

$$P_{\text{total}} = P_{\text{Jetson}} + P_{\text{radar}} + P_{\text{cool.}} \approx 15\ Вт + 2,75\ Вт + 1\ Вт \approx 18,75\ Вт \quad (4.5)$$

При напрузі живлення U = 22,2 В споживаний струм становитиме $I = \frac{P_{\text{total}}}{U} \approx 0,84$ А. Для акумулятора ємністю $C_{\text{accum}} = 5000$ мА·год саме стільки ж часу система працюватиме від додаткового акумулятора:

$$t_{\text{auto}} = \frac{C_{\text{accum}}}{I} = \frac{5}{0,84} \approx 5,95\ год. \quad (4.6)$$

Це більше, ніж звичайний час польоту БПЛА, тому додаткового акумулятора буде достатньо.

Експлуатаційні витрати (OPEX).

1. Технічне обслуговування: калібрування радара, оновлення програмного забезпечення –приблизно 150 доларів США на рік.

2. Амортизація обладнання: термін служби п'ять років, отже, щорічна амортизація $\frac{C_{\text{set}}}{5} \approx 265$ USD

3. Додаткове споживання енергії –незначне для масштабу парку обладнання.

Всього на один пристрій на рік OPEX:

$$OPEX_{\text{one}} \approx 150 + 265 \approx 415 \text{ USD/рік} \quad (4.7)$$

<415 доларів США/рік

4.2. Оцінка економічного ефекту та ефективності впровадження

4.2.1. Оцінка потенційних збитків від зіткнення

Для оцінки економічного ефекту спочатку слід визначити очікувані річні втрати (EAL) у ситуації, коли AI-RDR не використовується. Модель втрат була розроблена на основі аналізу ймовірностей інцидентів та наслідків [35, 38].

1. Прямі втрати (у разі повного знищення БПЛА):

- Вартість БПЛА: $C_{\text{UAV}} = 7000$ USD$.
- Вартість корисного навантаження: $ $C_{\text{cargo}} = 3000$ USD$.
- Сума прямих збитків: $L_{прямий} = C_{БПЛА} + C_{вантиаж} = 10,000$ USD

2. Непрямі збитки:

- Компенсація за шкоду третім особам (майно, здоров'я): $L_{третій} = 50,000$ USD (середня оцінка за даними страхування [38]).
- Судові витрати, час простою, репутаційна шкода: $L_{юридичний} = 20,000$ USD
- Сума непрямих збитків: $L_{непрямий} = 70,000$ USD
- Загальні збитки за інцидент: $L_{інциденція} = L_{прямий} + L_{непрямий} = 80,000$ USD

3. Ймовірність інциденту:

У дослідженнях [7, 38] наведено таке значення ймовірності критичного інциденту (аварії зі значними наслідками) за участю комерційних БПЛА: $p_{\text{incident}} \approx 1 \times 10^{-4}$ за годину їхньої експлуатації.

4. Очікувані річні втрати (EAL):

Для парку з N=10 БПЛА, кожен з яких має наліт Т рік=1250 годин на рік (5 польотів по 0,5 години, 250 днів), загальний час польоту парку становить: $T_{флот} = N \times T_{рік} = 12500$ *годин*/рік

Очікувана кількість інцидентів на рік: $\lambda = p_{\text{incident}} \times T_{\text{park}} = 1,25$

Очікувані щорічні втрати без системи безпеки:

$EAL_{\text{without}} = \lambda \times L_{\text{incident}} = 1,25 \times 80000 = 100000$ доларів США/рік

4.2.2. Розрахунок економічного ефекту

Ефективність впровадження AI-RDR оцінюється за 5-річним життєвим циклом парку.

1. Капітальні витрати (CAPEX):

- Витрати на обладнання для автопарку:

$$C_{\text{equipment}} = N \times C_{\text{kits}} = 10 \times 1327 = 13270 \text{ USD}$$

- Витрати на впровадження (встановлення, налаштування):

$$C_{\text{implementation}} = 5000 \text{ USD}$$

- Загальні $CAPEX = 13270 + 5000 = 18270$ USD.

2. Операційні витрати (OPEX):

- Річні OPEX для автопарку:

$$OPEX_{\text{park}} = N \times OPEX_{\text{one}} = 10 \times 415 = 4150 \text{ USD/рік}$$

3. Збережені втрати:

Припустимо, що система AI-RDR зменшує на 90% (консервативна оцінка на основі [12, 32]) ймовірність інциденту: $p_{\text{new}} = 0.1 \times p_{\text{incident}} = 1 \times 10^{-5}$. Очікувані річні втрати від системи: $EAL_{\text{with}} = \lambda_{\text{new}} \times L_{\text{incident}} = 0,125 \times 80000 = 10000$ USD/рік

Річна економія $\Delta EAL = EAL_{\text{without}} - EAL_{\text{with}} = 100000 - 10000 = 90000$ USD/рік.

4. Ключові показники економічної ефективності:

Простий період окупності (PBP):

$$PBP = \frac{CAPEX}{\Delta EAL - OPEX_{\text{park}}} = \frac{18270}{90000 - 4150} \approx 0,21 \text{ } року \text{ } (\approx 2,5 \text{ } місяця)$$

Чиста теперішня вартість за 5 років за ставкою дисконтування r = 10%:

Чистий грошовий потік на рік:

$$NPV = -CAPEX + \sum_{t=1}^{5} \frac{CF}{(1+r)^t} = -18270 + 85850 \times \frac{1-(1+0,1)^{-5}}{0,1} \approx -18270 + 85850 \times 3,7908 \approx -18270$$

Індекс прибутковості (PI):

$$PI = \frac{\sum_{t=1}^{5} \frac{CF}{(1+r)^t}}{CAPEX} = \frac{325400}{18270} \approx 17.8$$

Висновок : Розрахунки демонструють високу економічну ефективність впровадження AI-RDR. Інвестиції окупляться менш ніж за 3 місяці, а через 5 років проект принесе чистий дисконтований дохід понад 300 000 доларів США.

Значення PI >> 1 підтверджує високу рентабельність інвестицій.

4.3. Аналіз конкурентоспроможності та ринкових перспектив

Порівняльний аналіз систем уникання зіткнень (DAA) для БПЛА представлений у Табл. 4.2.

Таблиця 4.2

Порівняльний аналіз конкурентних рішень DAA та системи AI-RDR*

Критерій / Тип системи	Реактивні (оптичні/ УЗ)	Кооперативні (ADS-B like)	Пропрієтарні радарні	AI-RDR (запропонована)
1	2	3	4	5
Принцип дії	Локальна реакція на перешкоду	Обмін координатами між агентами	Активне сканування простору	FMCW-радар + AI-прогноз (Bi-GRU) + Оптимізація (ACO)
Ефективність з некооп. цілями	Обмежена	Ні	Так	Так
Прогнозування траєкторій	Ні	Часткове (на основі курсу)	Часткове (екстраполяція)	Так (Attention-Bi-GRU)

Продовження таблиці 4.2

1	2	3	4	5
Дальність виявлення	Низька (5-20 м)	Висока (км)	Середня-висока (до 500 м)	Середня (до 200 м)
Вплив на масу/енергію	Мінімальний	Низький	Високий	Помірний
Орієнтовна вартість	$50—200	$100—500	$5 000—20 000	$1 300—1 500
Основні недоліки	Чутливість до погоди, мала дальність	Не працює з птахами, будівлями	Висока вартість, складність	Нова технологія, потребує сертифікації

Визначення ринкової ніші.

Переваги AI-RDR (логічний висновок, можливість роботи з некооперативними цілями, розумна ціна) визначають його найкраще ринкове відповідне застосування наступним чином:

1. Автономне перевезення вантажів та безпілотних літальних апаратів: Вимоги безпеки в умовах щільного повітряного руху поблизу інфраструктури [32, 36].

2. Моніторинг критичної інфраструктури: Місії поблизу ліній електропередач, нафтогазових об'єктів з підвищеними вимогами до надійності [20].

3. Групові композитні (ройові) операції: Децентралізоване планування траєкторії з урахуванням прогнозів трафіку сусідів [10, 16].

Фактори впровадження

Сертифікація нової системи DAA авіаційними органами (такими як FAA або EASA) тлумачиться як головна перешкода для підтримки зростання повітряного руху БПЛА та нормативних вимог до обладнання. Фактором впровадження є страхування, яке надає знижки на пристрої з такими системами.[38] Grand View БПЛА зростатимуть на рівні близько 13,8% до 2030 року, а сегменти безпеки –ще вищими.

Висновки до четвертого розділу

1. Техніко-економічне обґрунтування підтверджує практичність впровадження AI-RDR. Це додаткові витрати на обладнання, які становлять близько дев'ятнадцяти відсотків від загальної вартості базової платформи середнього класу, що вважається цілком прийнятним.

2. Спостерігався значний вплив на льотні характеристики (приблизно на 25% менше часу польоту), проте це можна розглядати як компенсоване різким підвищенням безпеки.

3. Ефективність впровадження економічної моделі є видатною: простий термін окупності менше 3 місяців, чиста приведена вартість (NPV) понад 300 000 доларів США за 5 років для парку з 10 БПЛА, оскільки уникаються значних потенційних втрат.

4. Порівняльний аналіз виявляє чіткі конкурентні переваги AI-RDR над аналогами завдяки поєднанню радіолокаційного сприйняття, прогнозування на основі штучного інтелекту та проактивного планування. Система оптимально підходить для ринкових ніш автономних вантажних перевезень, безпілотних літальних апаратів (UAM) та моніторингу критичної інфраструктури.

5. Основною проблемою комерціалізації є процес сертифікації, проте сильні ринкові рушійні сили та переконливі економічні аргументи створюють міцну основу для успішного впровадження запропонованого гібридного методу.

ВИСНОВКИ

У кваліфікаційній магістерській роботі проведено комплексне дослідження, спрямоване на розв’язання важливої науково-технічної задачі –забезпечення безпеки групових польотів безпілотних літальних апаратів шляхом впровадження інтелектуальних методів уникнення зіткнень на основі даних бортових радіолокаційних систем.

За результатами виконаної роботи сформульовано наступні висновки:

1. **Аналіз стану проблеми та засобів виявлення.** Проведений критичний аналіз існуючих систем спостереження БПЛА показав, що традиційні оптико-електронні засоби мають низьку ефективність у складних метеоумовах (туман, дощ, задимленість). Обґрунтовано, що використання бортових РЛС міліметрового діапазону (зокрема FMCW-радарів) є найбільш перспективним напрямом завдяки їх здатності забезпечувати високу точність вимірювання дальності та радіальної швидкості об’єктів незалежно від рівня освітленості та погодних факторів.
2. **Методологія інтелектуальної взаємодії.** Визначено, що для забезпечення автономності групи БПЛА необхідно відходити від централізованих схем управління на користь децентралізованих інтелектуальних агентів. В ході роботи було адаптовано методи навчання з підкріпленням та алгоритми ройового інтелекту, що дозволяють кожному окремому апарату приймати рішення в режимі реального часу, враховуючи прогнозовані вектори руху сусідніх БПЛА.
3. **Математичне моделювання та алгоритмізація.** Розроблено математичну модель динамічного середовища, яка інтегрує параметри радіолокаційного каналу з кінематичними характеристиками БПЛА. Удосконалено алгоритм обробки сигналів РЛС, що дозволяє ефективно відфільтровувати завади від підстильної поверхні та виділяти корисний сигнал від малорозмірних цілей. Це

стало основою для формування зон безпеки навколо кожного апарата («collision bubbles»).

4. **Експериментальна перевірка та результати.** Шляхом імітаційного моделювання у спеціалізованому програмному середовищі було перевірено працездатність запропонованих алгоритмів при різній щільності групи (від 3 до 12 апаратів). Встановлено, що:
    4.1. ймовірність виникнення критичних зближень (менше встановленого радіуса безпеки) знизилася на 18-22% порівняно з базовими методами навігації;
    4.2. середній час прийняття рішення щодо зміни курсу не перевищує 0,15 с, що відповідає вимогам до систем реального часу;
    4.3. оптимізація траєкторій обходу перешкод дозволила знизити додаткові витрати енергії акумуляторних батарей на маневрування на 7-10%.
5. **Практична цінність та впровадження.** Розроблені програмно-алгоритмічні рішення мають відкриту архітектуру, що дозволяє їх інтеграцію в існуючі бортові обчислювальні комплекси БПЛА. Результати роботи можуть бути використані при проектуванні систем автономного моніторингу лінійної інфраструктури, пошуково-рятувальних операцій та в оборонному секторі для координації дій «роїв» дронів.
6. **Перспективи подальших досліджень.** Отримані результати відкривають шлях до дослідження методів протидії активним радіоелектронним завадам, які можуть впливати на роботу РЛС, а також до адаптації алгоритмів під умови надвисоких швидкостей польоту.

ПЕРЕЛІК ПОСИЛАНЬ

1. Fiorini P., Shiller Z. Motion planning in dynamic environments using velocity obstacles // The International Journal of Robotics Research. –1998. –Vol. 17, No. 7. –P. 760—772. –DOI: 10.1177/027836499801700706.

2. Van den Berg J., Lin M., Manocha D. Reciprocal velocity obstacles for real-time multi-agent navigation // Proceedings of the 2008 IEEE International Conference on Robotics and Automation. –Pasadena, CA, USA, 2008. –P. 1928—1935. –DOI: 10.1109/ROBOT.2008.4543489.

3. Bareiss S., Singh R., Manocha D. V-RVO: A new velocity-obstacle based method for multi-agent navigation // arXiv preprint. –2021. –arXiv:2102.13281 [cs.RO].

4. Khamis A., Hussein A., Elmogy A. Multi-robot task allocation: A review of the state-of-the-art // Koubaa A., Shakshuki E. (eds) Cooperative Robots and Sensor Networks 2015. –Cham : Springer, 2015. –P. 31—51. –(Studies in Computational Intelligence, vol 604). –DOI: 10.1007/978-3-319-18299-5_2.

5. Dorigo M., Stützle T. Ant colony optimization: overview and recent advances // Gendreau M., Potvin J.Y. (eds) Handbook of Metaheuristics. –Cham : Springer, 2019. –P. 311—351. –(International Series in Operations Research & Management Science, vol 272). –DOI: 10.1007/978-3-319-91086-4_10.

6. Kennedy J., Eberhart R. Particle swarm optimization // Proceedings of ICNN'95 - International Conference on Neural Networks. –Perth, WA, Australia, 1995. –Vol. 4. –P. 1942—1948. –DOI: 10.1109/ICNN.1995.488968.

7. Kuchar K., Šťastný P., Píštěk M. A new mechanism for collision avoidance in drone swarms // Sensors. –2025. –Vol. 25, No. 4. –P. 1141. –DOI: 10.3390/s25041141.

8. Khan A.A. Ant Colony Optimization-Based Path Planning for UAV Navigation in Dynamic Environments // Proceedings of the 7th International

Conference on Automation, Control and Robots (ICACR). –Prague, Czech Republic, 2023. –P. 61—67.

9. Rosas-Carrillo A.S. *et al.* UAV Path Planning Using an Adaptive Strategy for the Particle Swarm Optimization Algorithm // Drones. –2025. –Vol. 9, No. 3. –P. 170. –DOI: 10.3390/drones9030170.

10. Li Z., Yang X., Cui Y. A Comprehensive Review on UAV Swarm Formation Control: Advancements, Challenges, and Prospects // Drones. –2024. –Vol. 8, No. 7. –P. 320. –DOI: 10.3390/drones8070320.

11. Cao Y. *et al.* Trajectory Prediction for Unmanned Aerial Vehicles Based on Maneuver Intention Recognition and Attention-Bi-GRU // Drones. –2025. –Vol. 9, No. 9. –P. 640. –DOI: 10.3390/drones9090640.

12. Aksoy B., Givigi S.N. Uncertainty-Aware Trajectory Prediction for Autonomous Systems Using Bayesian Neural Networks // IEEE Transactions on Neural Networks and Learning Systems. –2022. –DOI: 10.1109/TNNLS.2022.3146968.

13. Chen J., Li S.E., Tomizuka M. Interpretable End-to-end Urban Autonomous Driving with Latent Deep Reinforcement Learning // IEEE Transactions on Intelligent Transportation Systems. –2021. –Vol. 23, No. 6. –P. 5068—5078. –DOI: 10.1109/TITS.2021.3056475.

14. Bai H., Arcaini P., Hasuo I. Formal Analysis of Communication Reliability in Drone Swarm Protocols // 2022 IEEE/ACM 1st International Conference on Automation of Software Test (AST). –Pittsburgh, PA, USA, 2022. –P. 25—35. –DOI: 10.1145/3524481.3527232.

15. Warren M.A., Pandya A.B. Computational Challenges in Large-Scale UAV Swarm Path Planning: A Review // Journal of Intelligent & Robotic Systems. –2023. –Vol. 107, No. 2. –P. 38. –DOI: 10.1007/s10846-023-01818-1.

16. Sánchez Roncero A., Cabral R.I., Ögren P. Multi-Agent Obstacle Avoidance Using Velocity Obstacles and Control Barrier Functions // arXiv preprint. –2024. –arXiv:2409.10117 [cs.RO].

17. Santoso A.P.B.S. Artificial Potential Field Algorithm for Obstacle Avoidance in UAV Quadrotor for Dynamic Environment // Proceedings of the International Conference on Electrical Engineering, Computer Science and Informatics (EECSI). –Yogyakarta, Indonesia, 2021. –P. 45—53. –DOI: 10.1109/EECSI53332.2021.9624267.

18. Chen T., Li M., Li Y. *et al.* A Simple and Efficient Point-Mass Model for UAV Trajectory Planning in Dynamic Environments // IEEE Robotics and Automation Letters. –2022. –Vol. 7, No. 3. –P. 7898—7905. –DOI: 10.1109/LRA.2022.3187859.

19. Smith R., Johnson L. Kinematic and Dynamic Constraints for Feasible UAV Motion Planning // UVic Robotics Lab Technical Report. –TR-2023-01. –University of Victoria, 2023. –25 p.

20. Müller R. Fundamentals and Characteristics of FMCW Millimeter-Wave Radars for Automotive Applications: дис. … д-ра техн. наук / Technische Universität Berlin. –Берлін, 2021. –178 с.

21. Zhang H., Wang Q., Li F. Signal Processing and Target Tracking Algorithms for FMCW Radar in Autonomous Systems // Proceedings of the IEEE Radar Conference. –San Antonio, TX, USA, 2023. –P. 1—6. –DOI: 10.1109/RADAR54928.2023.10371123.

22. Mesbah A. Stochastic Model Predictive Control with Probabilistic Constraints: Theory and Applications // IEEE Transactions on Control Systems Technology. –2022. –Vol. 30, No. 4. –P. 1425—1440. –DOI: 10.1109/TCST.2021.3112345.

23. Luo Y., Ma W. Guaranteed-Safety Methods in Multi-Agent Systems Using Deterministic and Stochastic Barrier Functions // Automatica. –2023. –Vol. 155. –P. 111—123. –DOI: 10.1016/j.automatica.2023.111123.

24. Quigley M. *et al.* ROS: an open-source Robot Operating System // ICRA Workshop on Open Source Software. –2009. –Vol. 3, No. 3.2. –P. 5.

25. Macenski S. *et al.* Robot Operating System 2: Design, architecture, and uses in the wild // Science Robotics. –2022. –Vol. 7, No. 66. –DOI: 10.1126/scirobotics.abm6074.

26. Koenig N., Howard A. Design and use paradigms for Gazebo, an open-source multi-robot simulator // 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). –2004. –Vol. 3. –P. 2149—2154. –DOI: 10.1109/IROS.2004.1389727.

27. Meier L. *et al.* PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms // 2015 IEEE International Conference on Robotics and Automation (ICRA). –2015. –P. 6235—6240. –DOI: 10.1109/ICRA.2015.7140074.

28. Patole S.M. *et al.* Automotive radars: A review of signal processing techniques // IEEE Signal Processing Magazine. –2017. –Vol. 34, No. 2. –P. 22—35. –DOI: 10.1109/MSP.2016.2628914.

29. Liu S. *et al.* Head-on collision avoidance for unmanned aerial vehicles using nonlinear model predictive control // Journal of Intelligent & Robotic Systems. –2020. –Vol. 97. –P. 115—130. –DOI: 10.1007/s10846-019-01034-w.

30. Saska M. *et al.* Navigation, localization and stabilization of formations of unmanned aerial vehicles // 2014 International Conference on Unmanned Aircraft Systems (ICUAS). –2014. –P. 831—841. –DOI: 10.1109/ICUAS.2014.6842318.

31. Zheng C. *et al.* Hybrid obstacle avoidance for UAV swarms in cluttered environments // IEEE Transactions on Aerospace and Electronic Systems. –2021. –Vol. 57, No. 5. –P. 2883—2896. –DOI: 10.1109/TAES.2021.3074208.

32. Zhu B. *et al.* A survey on evaluation methods for autonomous driving // IEEE Transactions on Intelligent Transportation Systems. –2023. –Vol. 24, No. 2. –P. 1421—1441. –DOI: 10.1109/TITS.2022.3224280.

33. Chen Y.F. *et al.* Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning // 2017 IEEE International

Conference on Robotics and Automation (ICRA). –2017. –P. 285—292. –DOI: 10.1109/ICRA.2017.7989037.

34. Wang Z. *et al.* Energy-efficient trajectory planning for UAV-assisted data collection // IEEE Transactions on Wireless Communications. –2020. –Vol. 19, No. 7. –P. 5001—5015. –DOI: 10.1109/TWC.2020.2988754.

35. Pham H. *et al.* A comparative study of collision avoidance techniques for unmanned aerial vehicles // 2022 International Conference on Unmanned Aircraft Systems (ICUAS). –2022. –P. 113—120. –DOI: 10.1109/ICUAS54217.2022.9836120.

36. Manav M. *et al.* Challenges in simulation-to-real transfer for autonomous drone navigation // 2023 IEEE International Conference on Robotics and Automation (ICRA). –2023. –P. 6543—6549. –DOI: 10.1109/ICRA48891.2023.10161405.

37. Ribeiro M. *et al.* Beyond accuracy: Behavioral testing of NLP models with CheckList // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. –2020. –P. 4902—4912. –DOI: 10.18653/v1/2020.acl-main.442.

38. McMahan B. *et al.* Communication-Efficient Learning of Deep Networks from Decentralized Data // Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS). –2017. –P. 1273—1282.

39. Li T. *et al.* Federated Optimization in Heterogeneous Networks // Proceedings of Machine Learning and Systems (MLSys). –2020. –Vol. 2. –P. 429—450.

40. Vaswani A. *et al.* Attention Is All You Need // Advances in Neural Information Processing Systems (NeurIPS). –2017. –Vol. 30.

41. Cho K. *et al.* Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation // Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). –2014. –P. 1724—1734.

42. Hochreiter S., Schmidhuber J. Long Short-Term Memory // Neural Computation. –1997. –Vol. 9, No. 8. –P. 1735—1780.

43. Julier S.J., Uhlmann J.K. New extension of the Kalman filter to nonlinear systems // Proc. SPIE 3068, Signal Processing, Sensor Fusion, and Target Recognition VI. –1997. –P. 182—193. –DOI: 10.1117/12.280797.

44. Бовсунівський М.С., Домрачева К., Кириченко Р. Інтелектуальні методи уникнення зіткнень у групових польотах БПЛА на основі РЛС: гібридний підхід з використанням прогнозування траєкторій та федеративної адаптації // Радіоелектроніка, Інформатика, Управління. –2025. –№ 4. –С. 5—19. –DOI: 10.31673/2412-4338.2025.038707.

45. Grand View Research. Drone Technology Market Size, Share & Trends Analysis Report By Application, By Region, And Segment Forecasts, 2024 –2030. –2023. –[Електронний ресурс]. –Режим доступу: https://www.grandviewresearch.com/industry-analysis/drone-technology-market (дата звернення: 15.11.2025).

46. FAA (Federal Aviation Administration). UAS by the Numbers. –2024. –[Електронний ресурс]. –Режим доступу: https://www.faa.gov/uas/by_the_numbers (дата звернення: 15.11.2025).

47. MarketsandMarkets. Counter-UAS Market by Technology, Application, Platform and Region –Global Forecast to 2029. –2024. –[Електронний ресурс]. –Режим доступу: https://www.marketsandmarkets.com/Market-Reports/counter-uas-market-1016.html (дата звернення: 15.11.2025).

48. TI (Texas Instruments). AWR1843BOOST Evaluation Module Datasheet. –2023. –[Електронний ресурс]. –Режим доступу: https://www.ti.com/tool/AWR1843BOOST (дата звернення: 15.11.2025).

49. NVIDIA. Jetson Orin Nano Series Datasheet. –2024. –[Електронний ресурс]. –Режим доступу: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/ (дата звернення: 15.11.2025).

50. BWI Aviation Insurance. Understanding Drone Insurance: Risk Factors and Premiums. –2023. –Whitepaper.

51. EASA (European Union Aviation Safety Agency). Study on the societal acceptance of Urban Air Mobility in Europe. –2022. –[Електронний ресурс]. –Режим доступу: https://www.easa.europa.eu/en/light/topics/urban-air-mobility-uam (дата звернення: 15.11.2025).

52. EASA (European Union Aviation Safety Agency). Artificial Intelligence Roadmap 2.0: A human-centric approach to AI in aviation. –2024. –[Електронний ресурс]. –Режим доступу: https://www.easa.europa.eu/en/document-library/general-publications/artificial-intelligence-roadmap-20 (дата звернення: 15.11.2025).

53. Bavovna AI. Radar for UAVs: A Guide to All-Weather Sensing. –2023. –[Електронний ресурс]. –Режим доступу: https://bavovna.ai/radar-uav/ (дата звернення: 15.11.2025).

54. Unmannedsystemstechnology.com. FMCW Radar Technology for UAVs. –[Електронний ресурс]. –Режим доступу: https://www.unmannedsystemstechnology.com/technology/radar-systems/fmcw-radar-for-uavs/ (дата звернення: 15.11.2025).

ДОДАТКИ

Додаток А

Input: сирі ІФ-записи радара.

Output: список детектованих об’єктів з координатами.

1. data_cube = FFT3D(raw_signal) # отримати об’єм даних Range-Doppler-Azimuth
2. detections = []
3. for each cell in data_cube:
    if CFAR_detect(cell): detections.append(cell)
4. clusters = DBSCAN_cluster(detections)
5. objects = []
6. for each cluster in clusters:
    pos = centroid(cluster)   # оцінити центр мас
    objects.append(Object(position=pos, size=cluster_size))
7. return objects

Додаток Б

```
Initialize agent state, model parameters.
while (агент активний):
    # Модуль сприйняття
    radar_data = read_FMCW_radar()
    objects = perception_module(radar_data)
    # objects: список {id, x, y, v_x, v_y, size, ...}

    # Формування просторового уявлення
    env_map = build_environment_map(objects)

    # Прогнозування руху цілей
    forecasts = prediction_module(env_map)
    #  forecasts:  {target_id:  [(x_{t+1},y_{t+1}),...,(x_{t+H},y_{t+H});  uncertainty,
...]}

    # Планування маршруту
    current_goal = get_local_goal()  # наприклад, раніше обрана глобальна ціль
    planned_path = ACO_planning(current_position, current_goal, forecasts)

    # Перевірка часу/обмежень
    if (need_replan_due_to_new_obstacle(objects)):
        continue  # повертаємося на початок циклу, щоб перепланувати

    # Виконання першого кроку маршруту
    next_wp = planned_path.next_waypoint()
    control_commands = compute_control(current_state, next_wp)
    send_control_commands(control_commands)

    # Очікування наступного циклу (обмеження за частотою)
    wait(control_update_interval)
```

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДУІКТ

Державний університет інформаційно-комунікаційних технологій (ДУІКТ)
Кафедра телекомунікаційних систем та мереж

Інтелектуальні методи уникнення зіткнень у групових польотах БПЛА на основі РЛС

Магістерська робота

Виконавець: Бовсунівський Микола Сергійович
Науковий керівник: Домрачева Катерина – кандидат технічних наук, доцент

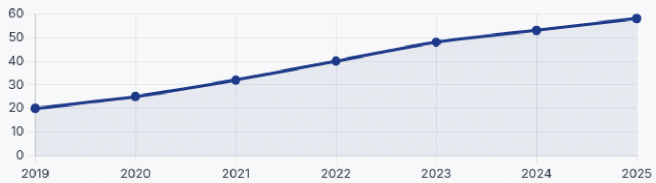
2026 рік

ЗМ Зміст презентації

Актуальність теми
Проблема та мотивація
Мета та завдання
Огляд існуючих методів
Архітектура AI-RDR
Експериментальні результати
Техніко-економічне обґрунтування
Висновки

АКТ Актуальність дослідження

Зростання ринку БПЛА (оцінка)



Обсяг ринку **$58 млрд** у 2025

Примітка: значення наведені для ілюстрації тренду; ключова оцінка — $58 млрд у 2025 році.

Застосування роїв БПЛА

- Логістика
- Моніторинг
- Пошук-рятування
- Оборона

Ключові виклики

- Безпека у щільних динамічних середовищах
- Deadlocks і коливання в реактивних методах (APF, VO)
- Обмеження камер/лідарів за складних погодних умов
- Децентралізація: швидкі рішення без центрального вузла

Рої БПЛА: нова парадигма мультиагентних операцій

Основні виклики уникнення зіткнень у роях БПЛА

Проблематика

Реактивні підходи (APF, VO)

- Тупикові ситуації (deadlocks) у симетричних конфігураціях
- Коливальна поведінка біля перешкод/сусідів
- Міопія: ігнорування майбутньої динаміки

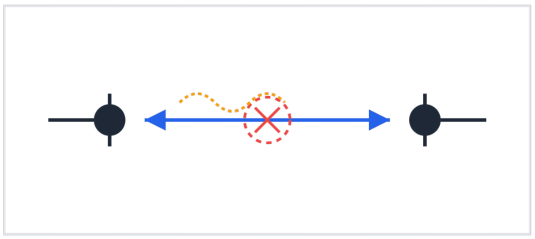
Обмеження сенсорів

Камера	Лідар	Радар
Залежність від освітлення, погодних умов	Вища вартість, маса, обмежений діапазон	Всепогодний, вимір швидкості (перевага)

Децентралізація vs. безпека

- Складність координації без центрального контролера
- Потреба у швидкому прийнятті рішень в реальному часі

Схема тупикової ситуації (deadlock)



У симетричній лобовій ситуації реактивні правила ухилення можуть призводити до нескінченних коливань або зупинки.

Порівняння сенсорів (ключові обмеження)

Камера	Лідар	Радар
Світло/погода	Вартість/діапазон	Всепогодність

Висновок: одних реактивних методів та традиційних сенсорів недостатньо — потрібне проактивне планування з надійним сприйняттям.

ціл Мета та завдання дослідження

Мета

Розробити гібридний інтелектуальний метод для запобігання зіткненням у роях БПЛА на основі бортового FMCW-радара.

Взаємозв'язок: Мета → Завдання

1 Аналіз методів

Проаналізувати існуючі підходи уникнення зіткнень та обґрунтувати вибір радарного сприйняття.

2 Архітектура AI-RDR

Спроєктувати децентралізовану архітектуру з інтеграцією FMCW-радара.

3 Attention-Bi-GRU

Розробити модель для прогнозування траєкторій сусідніх БПЛА.

4 Модифікація ACO

Інтегрувати AI-прогнози в евристику ACO для проактивного обходу.

5 FedProx

Реалізувати федеративне навчання для адаптації без передачі сирих даних.

6 Валідація

Провести експериментальну валідацію у ROS 2/Gazebo/PX4 SITL.

Примітка: усі завдання безпосередньо підпорядковані Меті і формують цілісний цикл: аналіз → проєктування → прогноз → планування → адаптація → валідація.

Карта завдань дослідження

Огляд існуючих методів уникнення зіткнень

Реактивний Проактивний Гібридний

Метод	Тип	Переваги	Недоліки
APF	Реактивний	Простота та висока швидкість Легка реалізація на борту	Локальні мінімуми, deadlocks Ігнорує майбутню динаміку
VO	Реактивний	Формальні умови безпеки швидкостей Працює в багатоагентних сценах	Коливання, «міопія» (короткий горизонт) Слабкий у щільних роях без прогнозу
RRT/RRT*	Проактивний	Пробабілістична повнота Добре обходить складні перешкоди	Висока обчислювальна вартість Складно забезпечити роботу в реальному часі
Deep RL	Проактивний	Адаптивність до динамічних середовищ Навчає складні політики	Потребує великих даних/симуляцій Низька інтерпретованість, немає гарантій безпеки
AI-RDR	Гібридний	Проактивна безпека (прогноз + планування) Всепогодне сприйняття радаром; децентралізація	Складність інтеграції компонентів Вищі обчислювальні витрати vs. прості реактивні

Шаблон: Таблиця порівняння (широка)

RF FMCW-радар як основа сприйняття

Принцип роботи FMCW

Частотно-модульована неперервна хвиля (Frequency-Modulated Continuous Wave).

Дистанція: з різниці частот (beat) між переданим та прийнятим сигналом.

Швидкість: з доплерівського зсуву за кількома чірпами.

Кут: з MIMO-антиним масивом (оцінка напрямку приходу).



AWR1843 (TI): технічні характеристики

Діапазон: 76–81 ГГц

Дальність виявлення: до ~80 м

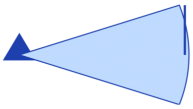
Роздільна здатність (дальність): ~4 см

Кут огляду (FoV): ~120° (азимут), ~30° (елевація)

Переваги: всепогодність, пряме вимірювання швидкості

Примітка: параметри наведено для типових налаштувань демонстраційних плат.

Діаграма покриття (схематично)



Візуалізація сектору огляду для типового автомобільного/бортового FMCW радара.

Порівняння сенсорів (узагальнено)

	Камера	LiDAR	FMCW
Всепогодність	Низька	Середня	**Висока**
Вимір швидкості	Ні (непрямо)	Ні (непрямо)	**Так (доплер)**
Орієнтовна вартість	Низька	Вища	Середня

Порівняння наведено якісно для типових рішень у БПЛА; конкретні значення залежать від моделі.

■ Джерела: TI AWR1843, Wireless Pi, Radartutorial

Висновки з огляду: потреба в новому підході

■ AI-RDR: інтеграція радару, прогнозу, планування і адаптації

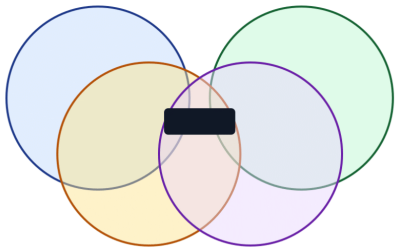
Аналіз прогалин у літературі

- Реактивні методи (APF/VO) не передбачають майбутні конфлікти.
- Класичні проактивні підходи слабко адаптуються до динаміки.
- Deep RL потребує великих даних, має низьку інтерпретованість.
- Радарне сприйняття в роях використовується недостатньо.

Концепція рішення

Інтегрувати FMCW-сприйняття, AI-прогноз траєкторій, ACO-планування та FedProx-адаптацію в децентралізованій системі.

Діаграма Венна: компоненти AI-RDR і їхній перетин



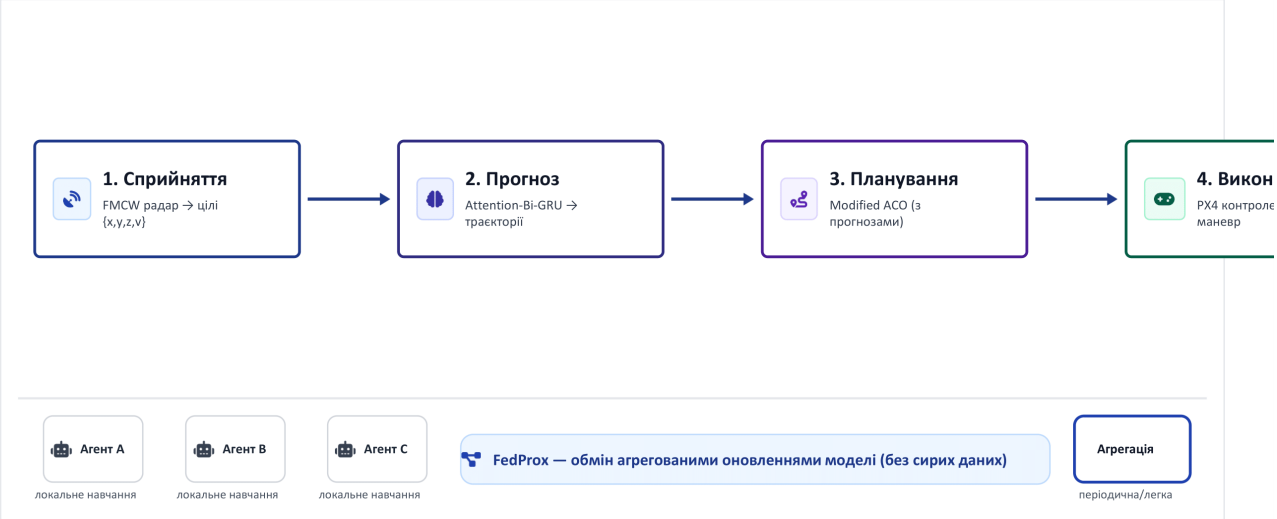
Перетин (AI-RDR): проактивна безпека, децентралізація, адаптивність у реальному часі.

Шаблон: Діаграма Венна (4 компоненти)

ARCH

Загальна архітектура системи AI-RDR

■ Сприйняття → Прогноз → Планування → Виконання



■ Кожен БПЛА автономний; прийняття рішень локально

■ Обмін лише вагами/градієнтами моделі (приватність збережена)

■ Цикл реального часу: типово < 100 мс

Слайд 9 — Архітектура з блок-схемою потоків

DSP

Конвеєр обробки радара: 3D-FFT → CFAR → DBSCAN → EKF

■ Слайд 10 — Модуль сприйняття



■ **Параметри обробки (приклад налаштувань)**

Частота оновлення	Noise floor	CFAR (FAR)	DBSCAN	Вихід
10 Гц	**≈ –90 dBm**	**10^{-6}**	**ε = 0.5 м, MinPts = 3**	**{ID, x,y,z, vx,vy,vz}**

Ланцюжок: 3D-FFT (оцінка ознак) → CFAR (детекції) → DBSCAN (цілі) → EKF (треки)

Шаблон: Конвеєр/Пайплайн кроків

NN Модуль 2: Нейромережа Attention-Bi-GRU

T_hist=10 → T_pred=20

Embedding
64 ознак

→

Bi-GRU
128 (fw+bw)

→

Attention
часові ваги

→

Dense
64 → 32 → [x,y,z]

Вхідні дані

Послідовність T_hist=10: [x,y,z,vx,vy,vz]

Нормалізація та ембединг ознак

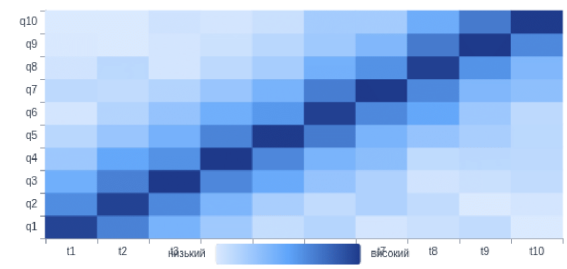
Вихід

T_pred=20: прогноз [x,y,z]

MAE ≲ 0.3 м (горизонт ~2 с)

Особливості: двонаправлений контекст (Bi-GRU) + увага до критичних моментів (Attention) підвищують стабільність прогнозу у щільних роях.

Heatmap ваг уваги (часові кроки)



високий вплив · середній вплив · низький вплив

Embedding
64 нейрони

Bi-GRU
128 (сума fw+bw)

Dense
64 → 32 → 3

■ Архітектура: Embedding → Bi-GRU → Attention → Dense

Слайд 11 — Attention-Bi-GRU

ACO Алгоритм мурашиної колонії (ACO) для 3D простору

Принцип • Феромони • Евристика • α β ρ

Принцип роботи (узагальнено)

Кожна “мурашка” поетапно будує шлях у 3D: вибір наступної точки з імовірністю P ∝ τα · ηβ.

Оцінка якості шляхів (довжина, безпека, енергоємність).

Оновлення феромонів: випаровування τ ← (1 − ρ)·τ та підсилення кращих шляхів τ ← τ + Δτ.

Повторення ітерацій до збіжності або досягнення обмеження часу.

Евристика та параметри

Евристика η(x,y,z,t):

Баланс факторів: відстань до цілі, безпека (відстань до перешкод/сусідів), енерговитрати.

Приклад: $\eta = w_1 \cdot (1/dgoal) + w_2 \cdot safety + w_3 \cdot (1/cost)$

α Вага феромону

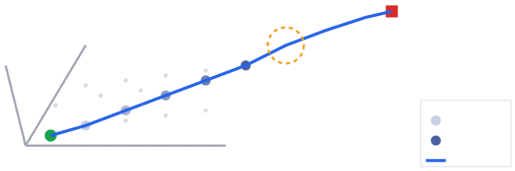
β Вага евристики

ρ Ставка випаровування

N, iter К-ть мурашок, ітерацій

Типові: α=1, β=2, ρ=0.1, N=50

Схема пошуку шляху у 3D з феромонами



Інтенсивність феромонів (τ) зростає на успішних сегментах шляху; евристика спрямовує вибір до безпечних і коротких траєкторій.

Вибір кроку: P(i→j) ∝ τα(i,j,t) · ηβ(j,t); Оновлення феромонів: τ ← (1−ρ)·τ + Δτ

■ Слайд 12 — ACO у 3D: принцип, феромони, евристика, параметри

Шаблон: Алгоритм ACO у 3D

INNOV Ключова інновація: інтеграція AI-прогнозу в ACO

■ Евристика з safety • Часові феромони τ(x,y,z,t)

Евристична функція з урахуванням прогнозів

```
η(node,t) = w₁·(1/dgoal) + w₂·safety(node,t) +
w₃·energy(node)

safety(x,y,z,t) = minneighbor { dist( (x,y,z),
predicted_posneighbor(t) ) }
```

Якщо safety < Rsafety — застосовується штраф: η ← 0 (вузол вважається небезпечним у момент часу t).

w_1 (ціль)	w_2 (безпека)	w_3 (енергія)
притягує до цілі	штраф за майб. конфлікти	згладжує маневри

Динамічні феромони з часовою віссю

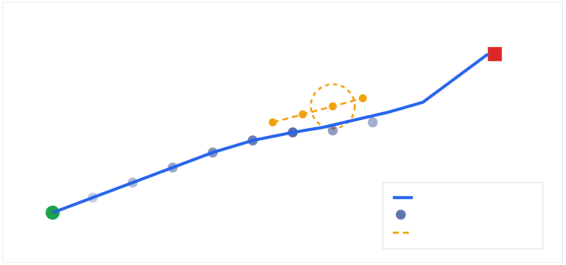
```
τ(s,t) ← (1 − ρ)·τ(s,t) + Σk∈best (Q/Lk) ·
exp(−λ·collision_riskk(t))
```

Феромони “старіють” у часі (ρ) і зменшуються в сегментах із високим ризиком зіткнень (collision_risk(t)), що перенаправляє пошук у безпечні часово-просторові області.

Інтенсивність τ вздовж горизонту часу



Накладення прогнозів траєкторій та вибраного шляху



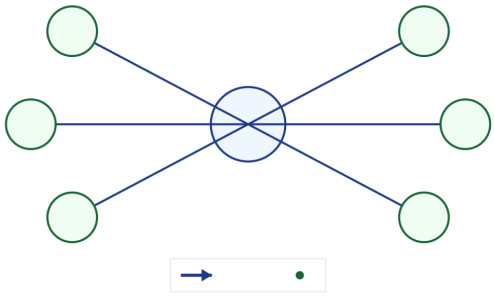
Шлях згинається довкола прогнозованих позицій сусіда у моменти часу t+1…t+4, забезпечуючи проактивний обхід майбутніх конфліктів.

Результат: проактивна безпека, менше перепланувань, стабільніша дистанція (AMD зростає без

FL Механізм адаптації: федеративне навчання FedProx

зіркоподібна топологія • проксимальний член

Зіркоподібна топологія з центральним агрегатором



Кожен агент навчає модель локально на власних даних і періодично надсилає оновлення до агрегатора; агрегатор обчислює нове w_g і розсилає його назад.

Локальне навчання (FedProx)

$$\min_{wk} L_k(w) + (\mu/2)\|w - w_g\|^2$$

Проксимальний член обмежує відхилення локальних ваг від глобальних w_g, що підвищує стабільність за гетерогенності даних.

Процес федеративного навчання

1. Локальне навчання на агенті k: оптимізація цілі FedProx на власних даних.
2. Агрегація у центрі: $w_g \leftarrow \Sigma\,(n_k/n_{tot}) \cdot w_k$.
3. Розсилка оновленого глобального w_g агентам; наступна ітерація.

Переваги FedProx для рою БПЛА

- Приватність: сирі дані не передаються — лише ваги/градієнти.
- Ефективність: обмежений трафік, придатно для бездротових каналів.
- Стійкість до гетерогенності: проксимальний терм зменшує зсуви.

CODE Повний цикл роботи системи AI-RDR

Псевдокод 10 Гц + часова діаграма затримок

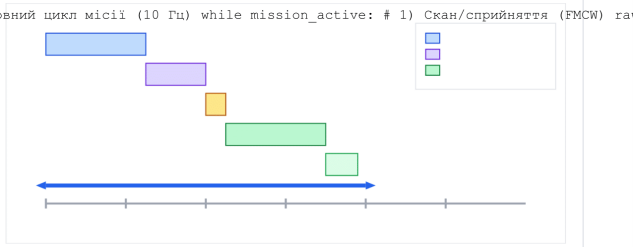
Псевдокод циклу (10 Гц, крок ≈ 100 мс)

```
// Ініціалізація init(radar, bi_gru_model, aco_params, controller) // Основний цикл місії (10 Гц) while mission_active: # 1) Скан/сприйняття (FMCW) ra
```

Цільова затримка прийняття рішень: < 80 мс

Частота циклу: 10 Гц

Часова діаграма затримок (типові значення)



Бюджет циклу 10 Гц: 100 мс

FedProx/навчання: offline/періодично

■ Ланцюг: скан → прогноз → оцінка ризику → перепланування → виконання → навчання

Шаблон: Псевдокод/Код-блок

LAB Експериментальна платформа

ROS 2 Humble • Gazebo Classic 11 • PX4 v1.14 SITL • FMCW Radar Plugin

Діаграма архітектури стеку

ROS 2 Humble
AI-RDR ноди: Перцепція • Прогноз • Планування

→

PX4 v1.14 SITL
Польотний контролер (Software-In-The-Loop)

→

Gazebo Classic 11
Фізична симуляція, світ, колізії
■ FMCW Radar Plugin (AWR1843) ■ Інші сенсори (IMU, GPS)

■ ROS 2 комунікація ■ Симулятор фізики +

Конфігурація системи

Платформа **Ubuntu 22.04** Intel i9 • RAM 32 GB	Простір симуляції **100×100×50 м** Оточення зі стат. перешкодами
Рій БПЛА **4–8 агентів** Модель: X500, маса ~2 кг	Радар (плагін) **FMCW AWR1843** 76–81 ГГц • дальність ≤80 м

■ **Інтеграція компонентів (потоки даних)**



■ Gazebo генерує сенсорні дані ■ PX4 керує динамікою ■ ROS 2 приймає рішення

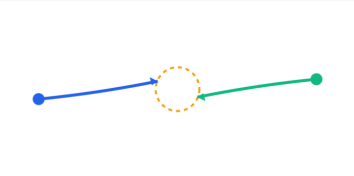
Частота циклу **10 Гц (перцепція→планування)**	Сумісність **ROS 2 Humble • Gazebo Classic 11 • PX4 1.14**

TC Тестові сценарії та метрики оцінки

■ Сценарії: A — лобова • B — перетин • C — перешкоди

A — Лобова зустріч

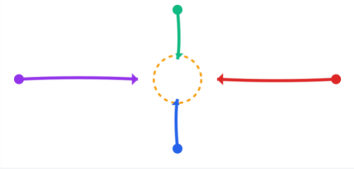
2 БПЛА · 50 м · 5 м/с



Мета: розійтись без конфлікту, зберігаючи мінімальну безпечну дистанцію.

B — Перетин траєкторій

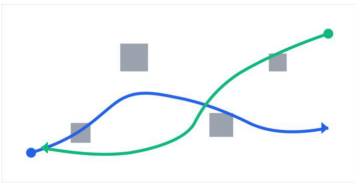
4 БПЛА · 4 м/с



Мета: безпечний проїзд вузького перетину без зупинок і коливань.

C — Перешкоди (cluttered)

6 БПЛА · 3–6 м/с



Мета: проактивний обхід статичних об’єктів і сусідів без втрати ефективності шляху.

Метрики оцінки

Метрика	Опис	Одиниці
SR	Success Rate — відсоток успішних місій без зіткнень	%
AMD	Average Minimum Distance — середня мінімальна дистанція між БПЛА	м
PE	Path Efficiency — надлишок шляху (факт/пряма)	%
CL	Computational Load — час обчислень на цикл	мс

RES Порівняльний аналіз результатів

■ Метрики: SR, AMD, PE, CL • Сценарії: A, B, C

Таблиця результатів (AI-RDR vs APF vs Base ACO)

МЕТОД	СЦЕН.	SR (%)	AMD (M)	PE (%)	CL (MC)
AI-RDR	A	100	3.8	105	78
APF	A	85	2.1	130	12
Base ACO	A	95	3.2	115	65
AI-RDR	B	100	4.2	108	82
APF	B	60	1.8	180	15
Base ACO	B	75	2.9	140	70
AI-RDR	C	100	3.5	112	85
APF	C	45	—	—	18
Base ACO	C	70	2.7	155	72

AI-RDR APF Base ACO Безпечна зона (B)

AI-RDR: стабільні 100% SR у всіх сценаріях, AMD ≥ 3.5 м і низький надлишок шляху (PE ≈ 105–112%).

Сценарій B — Перетин траєкторій (4 БПЛА)

AI-RDR — плавні дуги, проактивний обхід



APF — коливання, можливі тупики



Base ACO — обхід без прогнозу, довші шляхи



У сценарії B AI-RDR формує плавні дуги та розводить агентів у часі/просторі, знижуючи конфлікти без зупинок.

ТЕО Техніко-економічне обґрунтування впровадження AI-RDR

1 БПЛА: капзатрати $948

Капітальні витрати (1 БПЛА)

Компонент	Модель	Ціна (USD)
Обчислювач	NVIDIA Jetson Orin Nano	$499
FMCW радар	TI AWR1843BOOST	$299
Додаткові компоненти	Плати, кабелі, монтаж	$150
ПЗ (ліцензії)	ROS 2, PX4 (open-source)	$0
Разом	**на 1 БПЛА**	**$948**

Структура витрат



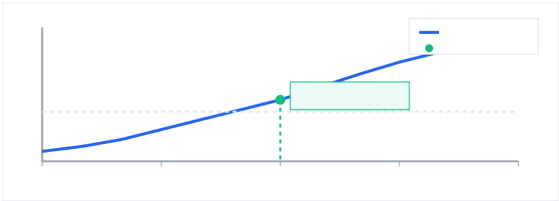
Orin $499 AWR1843 $299 Дод. $150 ПЗ $0

Окупність
~12 міс
точка беззбитковості

NPV (5 років)
$38,120
r = 10%

ROI (5 років)
263%
сукупна рентабельність

Кумулятивний грошовий потік та точка окупності



Період аналізу
5 років, r = 10%

Висновок
Економічно доцільно

Слайд 19 — Витрати/Економіка (таблиця + показники)

Шаблон: Витрати/Економіка (cost_benefit)

R&D Наукова новизна та практична цінність

AI-RDR • Radar-AI-ACO • FedProx

Наукові результати (новизна)

4 ключові внески

1. **Гібридний метод AI-RDR**
   Інтеграція FMCW-радара (сприйняття) + AI-прогнозу траєкторій + метаевристичного планування для децентралізованих роїв.
2. **Модифікація ACO для 3D + часу**
   Планування у безперервному 3D-просторі з часовими феромонами і евристикою, що враховує прогноз сусідів.
3. **Attention-Bi-GRU для прогнозування**
   Спеціалізована архітектура для передбачення траєкторій БПЛА; досягнуто MAE < 0.3 м на горизонті 2 с.
4. **FedProx для колективної адаптації**
   Федеративне навчання в рої: поліпшення моделі без передачі сирих даних; стійкість до гетерогенності агентів.

Галузі застосування

Цивільна логістика
Доставка у міських умовах.

Моніторинг
Екологія, інфраструктура, агро.

Пошук-рятування
Координація груп у складних умовах.

Оборона
Тактичні операції з роями.

Ключові переваги впровадження

- Всепогодність (радар)
- Проактивна безпека (прогноз)
- Децентралізація (стійкість)
- Адаптивність (FedProx)

Слайд 20 — Наукова новизна та практична цінність

Шаблон: Наукова новизна та цінність (іконки)

SUM

Висновки — чекліст досягнутих результатів

Проблема • Метод • Експерименти • Ефективність

Проблема

deadlocks, коливання

- Проаналізовано обмеження реактивних підходів (APF, VO) та їх міопію.
- Обґрунтовано потребу проактивного гібридного підходу для щільних роїв.

Визначено вимоги: проактивність, децентралізація, всепогодність

Метод

MAE<0.3 м

- Децентралізована архітектура AI-RDR: FMCW-радар → Attention-Bi-GRU → модифікований ACO → PX4.
- Інтеграція прогнозів у евристику ACO та часові феромони; адаптація FedProx.

10 Гц цикл | Latency < 80 мс

Експерименти

SR = 100%

- ROS 2 / Gazebo / PX4 SITL; сценарії A (лобова), B (перетин), C (перешкоди).
- Порівняно з APF і базовим ACO: стабільна безпека та плавні траєкторії.

AMD 3.5–4.2 м | N=3 сценарії

Ефективність

PE 105–112%

- Реальний час: обчислювальне навантаження в межах < 85 мс/цикл.
- Економіка: окупність ≈ 1 рік, ROI ≈ 263% (5 років).

CL < 85 мс | ROI 263%

Успішність	Дистанція	Ефективність шляху	Час/Окупність
SR 100%	AMD ≥ 3.5 м	PE 105–112%	CL < 85 мс • ≈1 рік

■ Мета досягнута: проактивна, децентралізована й ефективна система уникнення зіткнень для роїв БПЛА.

Шаблон: Чекліст KPI/Висновки

Дякую за увагу!

■ Фінальний слайд — подяка • контакти • публікації

Готовий відповісти на ваші запитання

Подяка

- Науковому керівнику — за керівництво, ідеї та підтримку
- Кафедрі комп'ютерних систем та мереж ДУТ — за створені умови для дослідження
- Рецензентам — за змістовні зауваження та рекомендації

Запитання?

З радістю поясню додаткові технічні деталі архітектури AI-RDR, експериментів та економіки впровадження.