

МАГІСТЕРСЬКА РОБОТА

МР . Шм – 00.00.000 ПЗ

Група Шм-24-1

Галас Володимир

2025

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Галас Володмир Богданович

(прізвище, ім'я, по-батькові)

УДК 004.8:656.07

(індекс)

МАГІСТЕРСЬКА РОБОТА

Методи оптимізації логістичних маршрутів із застосуванням AI-технологій

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 – Інженерія програмного забезпечення

(шифр і назва спеціальності)

Галас В. Б.

(підпис, ініціали та прізвище здобувача освітнього ступеня)

Науковий керівник **Козак Олексій Федорович, к.т.н., доцент**

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту

Завідувач кафедри

доц. Бандура В. В.

(посада) (підпис) (дата) (ініціали та прізвище)

Нормоконтроль

доц. Вовк Р. Б.

(посада) (підпис) (дата) (ініціали та прізвище)

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

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата
Перевірка на плагіат	доц., к.т.н. Вовк Р. Б.	

7. Дата видачі завдання 04 вересня 2025 р.

Керівник

_____ (підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назви етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір і вивчення літератури	01.10.2025	виконано
2	Виконання теоретичного дослідження	25.10.2025	виконано
3	Виконання аналітичного дослідження	10.11.2025	виконано
4	Обробка та аналіз результатів	22.11.2025	виконано
5	Підготовка пояснювальної записки	04.12.2025	виконано
6	Затвердження пояснювальної записки роботи завідувачем кафедри	15.12.2025	виконано

Студент – магістр

_____ (підпис)

Керівник роботи

_____ (підпис)

АНОТАЦІЯ

Магістерська робота: 82 с., 4 рис., 7 табл., 36 джерел, 2 додатки.

Тема: Методи оптимізації логістичних маршрутів із застосуванням AI-технологій.

Об'єкт дослідження: логістичні процеси маршрутизації в системах доставки.

Мета роботи: дослідження та аналіз методів оптимізації логістичних маршрутів на основі технологій штучного інтелекту.

Предмет дослідження: методи та алгоритми оптимізації логістичних маршрутів, зокрема генетичні алгоритми та їх використання у багатокритеріальних логістичних задачах.

Результати дослідження:

У магістерській роботі проаналізовано класичні та сучасні підходи до оптимізації логістичних маршрутів, виявлено їхні переваги та обмеження в умовах задач великої розмірності. Проведено огляд AI-методів, що застосовуються у логістиці, та обґрунтовано доцільність використання генетичного алгоритму як ефективного інструмента оптимізації. Розроблено формалізовану модель задачі маршрутизації, визначено основні критерії оптимізації та описано алгоритмічну послідовність реалізації генетичного алгоритму.

Висновок:

У результаті виконання роботи підтверджено ефективність застосування генетичного алгоритму для оптимізації логістичних маршрутів у задачах з великою кількістю обмежень та критеріїв.

ОПТИМІЗАЦІЯ ЛОГІСТИЧНИХ МАРШРУТІВ, ШТУЧНИЙ ІНТЕЛЕКТ, ГЕНЕТИЧНИЙ АЛГОРИТМ, БАГАТОКРИТЕРІАЛЬНА ОПТИМІЗАЦІЯ, ЛОГІСТИКА, МАРШРУТИЗАЦІЯ, ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ, СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ.

ABSTRACT

Master's thesis: 82 pages, 4 figures, 7 tables, 36 references, 2 appendices.

Title: Methods for Optimizing Logistics Routes Using AI Technologies.

Object of research: logistics routing processes in delivery systems.

Purpose of the thesis: to study and analyze methods for optimizing logistics routes based on artificial intelligence technologies.

Subject of research: methods and algorithms for optimizing logistics routes, in particular genetic algorithms and their application in multi-criteria logistics optimization problems.

Research results:

The master's thesis analyzes classical and modern approaches to logistics route optimization and identifies their advantages and limitations in large-scale problem settings. An overview of AI-based methods used in logistics is provided, and the feasibility of applying a genetic algorithm as an effective optimization tool is substantiated. A formalized model of the routing optimization problem is developed, key optimization criteria are defined, and the algorithmic sequence of the genetic algorithm implementation is described..

Conclusion:

As a result of the research, the effectiveness of using a genetic algorithm for optimizing logistics routes in problems with a large number of constraints and criteria has been confirmed.

LOGISTICS ROUTE OPTIMIZATION, ARTIFICIAL INTELLIGENCE, GENETIC ALGORITHM, MULTI-CRITERIA OPTIMIZATION, LOGISTICS, ROUTING, SOFTWARE ENGINEERING, DECISION SUPPORT SYSTEMS.

ЗМІСТ

ВСТУП.....	9
РОЗДІЛ 1	
АНАЛІЗ ЗАДАЧІ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ....	13
1.1 Задачі логістичної маршрутизації та їх формалізація	13
1.2. Класичні алгоритмічні методи оптимізації маршрутів	15
1.3. Обмеження класичних методів оптимізації маршрутів	19
1.4. Висновки до розділу	25
РОЗДІЛ 2	
МЕТОДИ ОПТИМІЗАЦІЇ МАРШРУТІВ ІЗ ЗАСТОСУВАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ	27
2.1. Огляд AI-методів у оптимізації маршрутів.....	27
2.2. Генетичні алгоритми у оптимізації логістичних маршрутів.....	28
2.3. Обґрунтування вибору генетичного алгоритму	37
2.4. Архітектурна організація та логіка поетапної реалізації генетичного алгоритму	41
2.5. Порівняння генетичного алгоритму з іншими AI-методами	43
2.6 Висновки до розділу	45
РОЗДІЛ 3	
ПОБУДОВА МЕТОДУ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ ІЗ ЗАСТОСУВАННЯМ ГЕНЕТИЧНОГО АЛГОРИТМУ	47
3.1. Загальна концепція методу оптимізації логістичних маршрутів	47
3.2 Критерії оптимізації та їх формалізація.....	49
3.3 Реалізація критеріїв оптимізації у структурі генетичного алгоритму	51
3.4 Послідовність роботи алгоритму оптимізації маршрутів	54
3.5. Висновки до розділу	56

РОЗДІЛ 4

ПРАКТИЧНА ІМПЛЕМЕНТАЦІЯ МЕТОДУ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ.....	58
4.1 Реалізація формування початкової популяції маршрутів.....	58
4.2 Аналіз результатів експериментального дослідження оптимізації логістичних маршрутів.....	63
4.3. Оціна повноти та достовірності результатів	65
4.4 Практична цінність та перспективи розвитку	68
4.5 Висновки до розділу	68
ВИСНОВКИ.....	70
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	72
ДОДАТКИ	76

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ЛЗ – логістична задача

ППЗ – інженерія програмного забезпечення

ІТ - інформаційні технології

ПЗ - програмне забезпечення

ГА - генетичний алгоритм

ЗОП - задача оптимального планування

АІ – штучний інтелект

ІоТ – інтернет речей

ВСТУП

Актуальність роботи

У сучасних умовах розвитку економіки логістика відіграє ключову роль у забезпеченні ефективного функціонування підприємств різних галузей. Оптимізація логістичних процесів, зокрема маршрутів перевезення, безпосередньо впливає на зниження витрат, підвищення швидкості доставки та покращення якості обслуговування споживачів. Ускладнення логістичних систем, зростання обсягів інформації та кількості обмежень зумовлюють необхідність застосування сучасних формальних і алгоритмічних методів аналізу та оптимізації.

Особливу актуальність у цьому контексті набувають методи комбінаторної оптимізації, зокрема метаевристичні підходи, які дозволяють отримувати ефективні рішення для складних задач маршрутизації. Серед таких методів важливе місце займають генетичні алгоритми, що ґрунтуються на принципах еволюційного пошуку та характеризуються гнучкістю, адаптивністю і можливістю застосування до задач великої розмірності. Саме ці властивості роблять їх доцільними для використання в логістичних системах.

Актуальність теми магістерської роботи зумовлена потребою в аналізі та обґрунтуванні ефективності формальних методів оптимізації логістичних маршрутів, а також у дослідженні можливостей їх застосування для розв'язання практичних задач маршрутизації. В умовах постійної зміни логістичних параметрів та обмежень важливим є не лише отримання оптимального рішення, а й розуміння принципів роботи алгоритмів, що використовуються для його пошуку.

Порівняння роботи з відомими розв'язаннями проблеми

У межах інженерії програмного забезпечення задача оптимізації логістичних маршрутів [1-22] традиційно реалізується з використанням класичних алгоритмів, таких як жадібні методи, динамічне програмування, повний перебір та локальні евристики. Такі підходи широко представлені у навчальних та прикладних програмних реалізаціях [23-25], проте їх використання обмежується задачами невеликої розмірності та статичними умовами.

На відміну від традиційних реалізацій, у даній роботі розглядається програмна реалізація методу оптимізації логістичних маршрутів із використанням генетичного алгоритму, який дозволяє ефективно працювати з великими наборами даних, підтримувати багатокритеріальну оптимізацію та легко інтегруватися у програмну архітектуру логістичних інформаційних систем. Запропонований підхід відповідає сучасним тенденціям розвитку інженерії програмного забезпечення, зокрема створенню модульних, масштабованих та інтелектуальних програмних продуктів.

Мета і задачі дослідження

Метою магістерської роботи є дослідження та аналіз методу оптимізації логістичних маршрутів на основі генетичного алгоритму, а також оцінка його ефективності для задач маршрутизації.

Досягнення мети включало розв'язання таких завдань:

- 1) аналіз теоретичних основ оптимізації логістичних маршрутів;
- 2) огляд сучасних підходів та методів розв'язання задач маршрутизації;
- 3) обґрунтування доцільності використання генетичних алгоритмів для оптимізації логістичних процесів;
- 4) побудова формалізованої моделі задачі оптимізації логістичних маршрутів.

Об'єктом дослідження є логістичні процеси маршрутизації в системах доставки.

Предметом дослідження є методи та алгоритми оптимізації логістичних маршрутів.

Методи дослідження

У процесі виконання магістерської роботи використовувався метод аналізу та синтезу для вивчення наукових джерел і узагальнення підходів до оптимізації логістичних маршрутів. Порівняльний аналіз застосовувався для зіставлення класичних алгоритмічних методів із AI-підходами та обґрунтування вибору генетичного алгоритму. Методи формалізації та математичного моделювання використовувалися для побудови моделі задачі оптимізації логістичних маршрутів з визначенням обмежень і критеріїв. Алгоритмічне моделювання застосовувалося для

розробки структури та логіки роботи генетичного алгоритму. Методи інженерії програмного забезпечення використовувалися під час проєктування та реалізації програмного рішення з урахуванням модульності та масштабованості.

Наукова новизна отриманих результатів

Досліджено можливості застосування методів штучного інтелекту для оптимізації логістичних маршрутів у програмних системах. Обґрунтовано доцільність використання генетичного алгоритму як ефективного алгоритмічного методу для розв'язання задач маршрутизації великої розмірності. Сформовано формалізовану модель задачі оптимізації логістичних маршрутів з урахуванням багатокритеріальних обмежень. Удосконалено алгоритмічну структуру розв'язання задачі маршрутизації шляхом застосування генетичного підходу в контексті інженерії програмного забезпечення. Удосконалено підхід до оптимізації логістичних маршрутів шляхом інтеграції багатокритеріальної функції пристосованості. Отримало подальший розвиток використання ГА в контексті програмної архітектури логістичних ІС

Практичне значення отриманих результатів

Запропонований алгоритм може бути використаний під час розробки програмного забезпечення логістичних інформаційних систем та інтелектуальних модулів оптимізації маршрутів, що забезпечує підвищення ефективності планування перевезень і скорочення витрат ресурсів. Результати дослідження можуть застосовуватися у системах підтримки прийняття рішень, а також у навчальному процесі при підготовці фахівців зі спеціальності «Інженерія програмного забезпечення» під час вивчення дисциплін, пов'язаних з алгоритмами оптимізації та штучним інтелектом.

Особистий внесок студента

Основним результатом є:

1. запропонований автором підхід до оптимізації логістичних маршрутів із використанням методів штучного інтелекту на основі генетичного алгоритму;
2. описана реалізація алгоритмічної моделі генетичного алгоритму, що виконує оптимізацію логістичних маршрутів з урахуванням багатокритеріальних обмежень та використовується у програмних логістичних системах.

Апробація результатів магістерської роботи

Результати магістерського дослідження були представлені та схвалені на науковій конференції інституту інформаційних технологій ІФНТУНГ.

Структура та обсяг магістерської роботи

Магістерська робота викладена на 85 сторінках друкованого тексту, який складається із вступу, чотирьох розділів, висновків, списку використаних джерел (36 найменувань). Робота містить 7 таблиць, 4 рисунки та 2 додатки, обсягом 4 стор.

РОЗДІЛ 1

АНАЛІЗ ЗАДАЧІ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ

У наукових дослідженнях проблема оптимізації логістичних маршрутів формувалася поступово, від класичних математичних моделей до сучасних інтелектуальних методів. Перші підходи ґрунтувалися на формалізації задач комбінаторної оптимізації, зокрема задачі комівояжера та її узагальнень, що дозволило закласти теоретичну основу маршрутизації [1-10]. Подальший розвиток отримали точні алгоритмічні методи, такі як повний перебір та динамічне програмування, які забезпечують знаходження оптимального розв'язку, проте характеризуються експоненційною обчислювальною складністю [22-26].

Зі зростанням розмірності реальних логістичних систем дослідження змістилися у бік евристичних і локальних методів пошуку, здатних знаходити наближені рішення за прийнятний час [30-35]. Наступним етапом розвитку стали метаевристичні підходи, що імітують природні процеси, зокрема еволюційні алгоритми. У сучасних роботах [1] все більшої уваги набувають методи штучного інтелекту, які поєднують здатність до адаптації, масштабованість та можливість роботи в динамічних умовах, характерних для сучасних логістичних систем.

1.1 Задачі логістичної маршрутизації та їх формалізація

Оптимізація логістичних маршрутів є однією з ключових задач у сфері транспортної та складської логістики, безпосередньо впливаючи на ефективність функціонування логістичних систем, рівень витрат і якість обслуговування клієнтів. У практичній діяльності такі задачі виникають під час планування доставки товарів, організації маршрутів транспортних засобів, розподілу ресурсів та мінімізації часових і фінансових витрат.

З формальної точки зору більшість задач логістичної маршрутизації зводяться до класу комбінаторних оптимізаційних задач, для яких характерна велика кількість можливих рішень та експоненційне зростання складності зі збільшенням розмірності

вхідних даних. Однією з базових моделей є задача комівояжера (Travelling Salesman Problem, TSP), яка полягає у знаходженні найкоротшого маршруту проходження через задану множину пунктів із поверненням до початкової точки. Незважаючи на простоту формулювання, задача TSP належить до класу NP-складних і має значну практичну цінність як основа для більш складних логістичних моделей.

Більш узагальненою та прикладно орієнтованою є задача маршрутизації транспортних засобів (Vehicle Routing Problem, VRP). Вона передбачає визначення оптимальних маршрутів для множини транспортних засобів, які здійснюють доставку або обслуговування клієнтів з одного або кількох депо з урахуванням обмежень на вантажопідйомність, кількість транспортних засобів, часові вікна, відстань або тривалість маршруту. У реальних логістичних системах VRP має численні модифікації, зокрема VRP з часовими вікнами, VRP з багатьма депо, VRP зі зворотною доставкою, що значно ускладнює процес пошуку оптимального рішення.

Формалізація задачі оптимізації логістичних маршрутів зазвичай включає визначення множини вузлів (пунктів доставки або обслуговування), матриці відстаней або часу між ними, набору обмежень та цільової функції. Як правило, цільова функція спрямована на мінімізацію загальної довжини маршрутів, сумарного часу виконання доставки або логістичних витрат. При цьому необхідно забезпечити виконання всіх заданих обмежень, що робить задачу багатокритеріальною та складною для точного розв'язання класичними методами.

Через NP-складність задач TSP і VRP використання точних алгоритмів є доцільним лише для задач невеликого розміру. У випадку реальних логістичних систем, де кількість пунктів доставки може сягати сотень або тисяч, традиційні алгоритмічні підходи стають неефективними. Це зумовлює необхідність застосування наближених, евристичних та метаевристичних методів, серед яких важливе місце посідають алгоритми штучного інтелекту, зокрема генетичні алгоритми.

Таким чином, задачі логістичної маршрутизації є складними комбінаторними задачами оптимізації, формалізація яких створює основу для подальшого аналізу

методів їх розв'язання та обґрунтовує доцільність застосування AI-підходів у сучасних програмних логістичних системах.

1.2. Класичні алгоритмічні методи оптимізації маршрутів

Оптимізація логістичних маршрутів є однією з центральних задач у транспортній та складській логістиці. Класичні алгоритмічні методи забезпечують детерміновані рішення, що дозволяє планувати маршрути та обчислювати оптимальні або наближені шляхи для транспортних засобів. Однак у задачах великої розмірності ефективність традиційних алгоритмів значно знижується

1.2.1. Жадібні алгоритми (*Greedy Algorithms*)

Жадібні алгоритми є одними з найпростіших і найпоширеніших методів оптимізації маршрутів. Принцип їхньої роботи полягає у послідовному виборі локально оптимального рішення на кожному кроці. Для задачі комівояжера (TSP) жадібний алгоритм обирає наступну вершину маршруту, яка має мінімальну відстань від поточного вузла, і продовжує процес до повернення у вихідну точку.

Формально, якщо (V) — множина вузлів, а $(d_{i,j})$ — відстань між вузлами (i) і (j) , жадібний алгоритм вибирає наступний вузол (k) за правилом:

$$k = \arg_{j \in V'} \min d(i, j) \quad (1.1)$$

де (V') — множина ще не відвіданих вузлів, (i) — поточний вузол маршруту.

До прикладу, якщо є 5 пунктів доставки з відстанями:

$$A \rightarrow B = 10, A \rightarrow C = 15, A \rightarrow D = 20, A \rightarrow E = 25, \quad (1.2)$$

жадібний алгоритм вибере спочатку вузол B, потім найближчий від B, і так далі.

Переваги цього методу полягають у простоті реалізації та низькій обчислювальній складності. Недоліком є те, що він не гарантує глобально оптимальний маршрут, а лише локально оптимальний, що для великих систем може призводити до значного збільшення загальної довжини маршруту.

1.2.2. Повний перебір (*Brute-Force Search*)

Метод повного перебору передбачає оцінку всіх можливих комбінацій маршрутів та вибір оптимального варіанту. Для задачі з (n) вузлами кількість можливих маршрутів дорівнює $((n - 1)!)$, що зростає експоненційно зі збільшенням розмірності задачі (1.3):

$$K = (n - 1)! \quad (1.3)$$

Цей метод забезпечує точне рішення, проте є практично непридатним для великих мереж доставки. Наприклад, для 10 пунктів доставки необхідно перевірити $(9! = 362.880)$ маршрутів, а для 20 пунктів — вже $(19! \approx 1.216 \times 10^{17})$ варіантів, що є обчислювально непосильним для сучасних комп'ютерів.

Повний перебір доцільний лише для навчальних або демонстраційних задач невеликого розміру та використовується як еталон для порівняння з іншими алгоритмами.

1.2.3. Динамічне програмування

Динамічне програмування дозволяє розбити задачу на підзадачі та обчислити оптимальні маршрути на основі рекурсивних співвідношень. Для задачі TSP застосовується алгоритм Беллмана–Хелда–Карпа, який формалізується таким чином:

$$C(S, j) = \min \left\{ C\left(\frac{S}{\{i\}}, i\right) + d(i, j) \right\} \quad (1.4)$$

де S — множина відвіданих вузлів, j — поточний вузол, $C(S, j)$ — мінімальна вартість маршруту через підмножину (S), що закінчується у вузлі (j).

Цей метод забезпечує точне рішення для задач середньої розмірності та широко застосовується для TSP до 20–25 вузлів. Його перевагою є точність і детермінованість результату, проте складність зберігання всіх підзадач зростає експоненційно, що обмежує практичне використання для великих логістичних систем.

1.2.4. Локальний пошук та оптимізаційні евристику (2-opt, 3-opt)

Методи локального пошуку покращують існуючий маршрут шляхом послідовної оптимізації окремих сегментів. Один з класичних методів — 2-opt: перестановка двох сегментів маршруту для зменшення загальної довжини шляху.

Якщо маршрут представлений послідовністю вузлів $[v_1, v_2, \dots, v_n]$, алгоритм 2-opt вибирає дві ребра (v_i, v_{i+1}) та (v_j, v_{j+1}) і міняє їх місцями, якщо нова довжина маршруту менша:

$$\Delta = d(v_i, v_j) + d(v_{i+1}, v_{j+1}) - d(v_i, v_{i+1}) - d(v_j, v_{j+1}) \quad (1.5)$$

Якщо $\Delta < 0$, відбувається заміна.

Методи 3-opt та k-opt виконують аналогічні перестановки трьох або k сегментів маршруту. Локальні евристику дозволяють значно покращити результат, отриманий жадібними або випадковими алгоритмами, і є ефективними для задач середньої розмірності (10–50 вузлів).

Зважаючи на представлену інформацію, кожен із методів володіє рядом переваг та недоліків за критеріями, які зведені в таблицю 1.1

Незважаючи на теоретичну обґрунтованість класичних алгоритмічних методів, їх практичне застосування у сучасних логістичних системах є суттєво обмеженим. Основною проблемою є експоненційне зростання обчислювальної

складності зі збільшенням кількості пунктів маршруту, що унеможливорює використання точних алгоритмів у системах реального масштабу.

Таблиця 1.1

Порівняння класичних методів

Метод	Складність	Точність	Масштабованість	Переваги	Недоліки
Жадібний алгоритм	$O(n^2)$	Локальна	Висока	Простота, швидкість	Не гарантує глобальний оптимум
Повний перебір	$O((n-1)!)$	Глобальна	Низька	Точність	Експоненційний ріст складності
Динамічне програмування	$O(n^2 \cdot 2^n)$	Глобальна	Середня	Точне рішення для середніх n	Обмеження на великий n
Локальний пошук (2-орт)	$O(n^2) - O(n^3)$	Наближена	Висока	Покращує існуючий маршрут	Може залишитись локальний мінімум

Крім того, більшість класичних методів орієнтовані на оптимізацію за одним критерієм та працюють у статичних умовах, що не відповідає вимогам сучасних програмних логістичних систем. У контексті інженерії програмного забезпечення це ускладнює інтеграцію таких алгоритмів у масштабовані інформаційні системи, де необхідна адаптація до змінних вхідних даних, обмежень і вимог користувачів.

Класичні методи є ефективними для задач невеликого розміру або навчальних прикладів. У реальних логістичних системах з десятками та сотнями пунктів доставка стає нерозв'язною з точністю без застосування наближених чи AI-методів.

Класичні алгоритмічні методи оптимізації маршрутів забезпечують важливу основу для розуміння логістичних задач. Жадібні алгоритми, повний перебір, динамічне програмування та локальні евристики мають свої переваги та обмеження. Для задач невеликого розміру ці методи забезпечують точні або близькі до оптимальних рішення, але для великих та складних логістичних систем вони виявляються недостатніми. Це створює передумови для застосування методів штучного інтелекту, здатних ефективно знаходити наближені оптимальні рішення у складних умовах, що буде розглянуто у наступному підрозділі.

1.3. Обмеження класичних методів оптимізації маршрутів

Класичні алгоритмічні методи оптимізації маршрутів відіграли важливу роль у становленні теорії комбінаторної оптимізації та логістичної маршрутизації. Вони широко застосовувалися для розв'язання задач комівояжера та маршрутизації транспортних засобів на початкових етапах розвитку логістичних інформаційних систем. Проте зі зростанням розмірності реальних задач і ускладненням умов доставки стало очевидним, що можливості таких методів є обмеженими.

Одне з фундаментальних обмежень класичних методів пов'язане з експоненційним зростанням обчислювальної складності. Р. Беллман у межах теорії динамічного програмування [22] звернув увагу на явище «прокляття розмірності», яке полягає у різкому зростанні кількості можливих станів зі збільшенням числа вузлів маршруту. Подальші дослідження в галузі теорії алгоритмів [36] формально довели NP-складність задач TSP і VRP, що означає відсутність ефективних поліноміальних алгоритмів їх точного розв'язання.

Іншою суттєвою проблемою є обмежена здатність класичних евристичних методів знаходити глобально оптимальні рішення. У працях [29,30] було показано, що методи локального пошуку ефективно покращують початкові маршрути, однак часто зупиняються у локальних мінімумах. Для великих логістичних систем це може призводити до накопичення неефективних маршрутних рішень і зростання сумарних витрат.

Значна увага у наукових дослідженнях приділялася також питанню масштабованості класичних алгоритмів [23,35]. У роботах, присвячених задачам маршрутизації транспортних засобів, [26,27] було доведено, що алгоритми, ефективні для десятків пунктів доставки, стають практично непридатними при роботі з сотнями вузлів. Час виконання таких методів різко зростає, а якість рішень перестає бути стабільною.

Окремим напрямом досліджень стали обмеження класичних методів у динамічних умовах [1]. Традиційні алгоритми проектувалися для статичних задач із

фіксованими параметрами, що не відповідає вимогам сучасної логістики. Зазначалося [29] , що будь-яка зміна вхідних даних потребує повного перерахунку маршруту, що суттєво знижує оперативність систем планування доставки.

Таблиця 1.2

Обмеження класичних методів оптимізації маршрутів

Показник / Обмеження	Опис	Приклад або пояснення
Експоненційне зростання складності	Обчислювальні ресурси зростають експоненційно зі збільшенням числа вузлів	Повний перебір для 20 пунктів маршрутів — непрактично для комп'ютера
Локальна оптимізація	Алгоритми часто знаходять локальний мінімум, а не глобальний	2-opt може покращити маршрут, але залишає неефективні сегменти маршруту
Недостатня гнучкість	Статичні алгоритми не адаптуються до динамічних змін	Затримки транспорту, зміни замовлень або погодні умови потребують повного повторного розрахунку
Обмежена масштабованість	Методи втрачають ефективність при великій кількості вузлів	Локальні евристичні методи ефективні для 10–50 пунктів, а для сотень вузлів час виконання стає непередбачуваним
Багатокритеріальна оптимізація	Традиційні методи орієнтовані на один критерій (відстань або час)	Сучасна логістика потребує одночасної оптимізації відстані, часу доставки, витрат пального, екології
Потреба у великих обчислювальних ресурсах	Точні алгоритми потребують значних обчислень для великих задач	Динамічне програмування: складність ($O(n^2 \cdot 2^n)$) для TSP, обмежує практичне використання

Крім того, класичні методи оптимізації здебільшого орієнтовані на однокритеріальні задачі, найчастіше — мінімізацію довжини маршруту. Сучасні логістичні системи вимагають одночасного врахування часу доставки, витрат пального, вартості перевезень та екологічних показників. Інтеграція таких критеріїв

у межах класичних алгоритмів призводить до значного ускладнення моделей і подальшого зростання обчислювальної складності.

Узагальнені обмеження класичних методів оптимізації маршрутів наведено в таблиці 1.2, яка систематизує основні проблеми їх практичного застосування в логістичних системах.

Таким чином, сучасні прикладні роботи підтверджують, що класичні алгоритмічні методи є ефективними лише для задач невеликого масштабу. Виявлені обмеження — експоненційна складність, локальна оптимізація, низька адаптивність і слабка масштабованість — обґрунтовують необхідність застосування більш гнучких підходів, зокрема методів штучного інтелекту. Розглянемо ці передумови детальніше.

Класичні алгоритми також недостатньо гнучкі у динамічних умовах. Вони працюють із фіксованими даними та припускають сталі відстані між вузлами, сталі умови задачі та наявність повної інформації про маршрути. У реальних системах логістики часто виникають непередбачувані події, такі як затримки транспорту, зміни замовлень у режимі реального часу або погодні умови, що вимагають адаптації маршрутів. Класичні методи не здатні швидко реагувати на такі зміни, і кожна модифікація маршруту потребує повторного розрахунку, що значно підвищує обчислювальні витрати та знижує ефективність системи.

Ще одним суттєвим обмеженням є низька масштабованість для великих логістичних мереж. Методи, ефективні для 10–50 пунктів, втрачають продуктивність при збільшенні кількості вузлів. Кількість перестановок та комбінацій, які потрібно перевірити для забезпечення оптимального або наближеного рішення, зростає експоненційно, що призводить до непередбачуваного часу виконання та нестабільної якості маршрутів.

Класичні алгоритми також обмежені у можливостях багатокритеріальної оптимізації. Сучасні логістичні задачі потребують врахування одночасно кількох критеріїв: довжини маршруту, часу доставки, вартості перевезення, витрат пального, екологічних показників тощо.

Традиційні алгоритми, орієнтовані на одномірну оптимізацію, наприклад мінімізацію відстані, не забезпечують ефективної інтеграції декількох критеріїв без значного ускладнення алгоритму та підвищення обчислювальної складності.

Тому класичні методи оптимізації маршрутів, хоча й забезпечують детерміновані та точні результати для невеликих задач, мають низку суттєвих обмежень: експоненційне зростання обчислювальної складності, ризик локальної оптимізації замість глобальної, недостатню гнучкість при динамічних змінах, обмежену масштабованість та складнощі з багатокритеріальною оптимізацією.

Ці фактори роблять традиційні методи недостатніми для реальних логістичних систем, особливо великих і динамічних.

Саме ці обмеження створюють передумови для використання методів штучного інтелекту, які дозволяють знаходити наближені оптимальні рішення та ефективно адаптуватися до змінних умов.

Обмеження класичних алгоритмічних методів (табл. 1.2) у логістичних системах створюють необхідність пошуку більш гнучких та масштабованих рішень. Методи штучного інтелекту (AI) стають оптимальним вибором для вирішення складних задач маршрутизації, оскільки вони дозволяють ефективно працювати з великими обсягами даних, адаптуватися до змінних умов і забезпечувати наближені оптимальні рішення навіть у випадку NP-складних задач.

Однією з головних передумов застосування AI є потреба у швидкому отриманні рішення для великих систем доставки

. Класичні методи, як показано раніше, не здатні обробляти сотні або тисячі пунктів у розумні терміни. Алгоритми штучного інтелекту, такі як генетичні алгоритми, алгоритми рою частинок, алгоритми імітації відпалу та нейронні мережі, дозволяють знаходити наближені оптимальні маршрути за прийнятний час, що робить їх ефективними для великих та динамічних систем.

Другою передумовою є необхідність адаптації до динамічних змін у маршрутах. У сучасній логістиці часто виникають непередбачувані події, такі як зміни замовлень, затримки транспорту, погодні умови або аварійні ситуації. AI-

алгоритми здатні швидко оновлювати маршрути на основі нових даних без повного повторного розрахунку, використовуючи евристичні та метаевристичні підходи. Наприклад, генетичні алгоритми дозволяють змінювати частину хромосом, що представляють маршрут, та отримувати оновлене наближене рішення з мінімальними витратами часу.

Третьою передумовою є необхідність багатокритеріальної оптимізації. Логістичні маршрути часто оптимізуються за кількома критеріями одночасно: мінімізація відстані, часу доставки, витрат пального, зменшення викидів CO₂ та дотримання часових вікон. AI-алгоритми здатні інтегрувати ці критерії у єдину цільову функцію або застосовувати багатокритеріальні методи оптимізації, забезпечуючи баланс між різними параметрами та досягаючи прийняттого компромісу.

Четвертою передумовою є необхідність гнучкого моделювання складних обмежень. У реальних логістичних системах маршрутизація має враховувати обмеження на вантажопідйомність транспортних засобів, часові вікна обслуговування, можливість зворотних доставок та інAI специфічні умови. AI-алгоритми дозволяють ефективно інтегрувати ці обмеження у процес оптимізації, наприклад шляхом додавання штрафних функцій у цільову функцію або використання спеціалізованих евристик.

П'ятою передумовою є можливість інтеграції з сучасними інформаційними системами та потоками даних. AI-алгоритми можна поєднувати з системами GPS, ERP та IoT, що дозволяє отримувати актуальні дані у режимі реального часу та автоматично коригувати маршрути. Це забезпечує високу гнучкість і надійність логістичних процесів навіть у складних умовах.

Отже, передумови застосування штучного інтелекту у оптимізації логістичних маршрутів (рис. 1.1) впливають із обмежень класичних методів та сучасних вимог до логістики: великий масштаб задач, динамічність середовища, багатокритеріальність, складність обмежень і потреба у інтеграції з інформаційними системами. Саме ці фактори визначають доцільність використання AI-підходів, серед

яких генетичні алгоритми посідають особливе місце завдяки своїй ефективності, гнучкості та здатності швидко знаходити наближені оптимальні рішення для складних задач маршрутизації.



Рис. 1.1 Передумови застосування AI

Класичні алгоритмічні методи оптимізації маршрутів, включаючи жадібні алгоритми, повний перебір, динамічне програмування та локальні евристики, ефективні для задач невеликого масштабу, забезпечують детерміновані результати та простоту реалізації, проте мають суттєві обмеження для великих і динамічних логістичних систем.

Основними обмеженнями класичних методів є експоненційне зростання обчислювальної складності зі збільшенням числа вузлів, ризик знаходження локальних мінімумів замість глобального оптимуму, недостатня гнучкість при змінах умов, обмежена масштабованість та складнощі з багатокритеріальною оптимізацією.

Виявлені обмеження створюють потребу у використанні більш гнучких і ефективних підходів, здатних швидко знаходити наближені оптимальні рішення, адаптуватися до змінних умов і враховувати декілька критеріїв одночасно.

Методи штучного інтелекту, зокрема генетичні алгоритми, алгоритми рою частинок та нейронні мережі, можуть успішно компенсувати обмеження класичних

алгоритмів завдяки високій масштабованості, здатності до адаптації в режимі реального часу, багатокритеріальній оптимізації та ефективному використанню обчислювальних ресурсів.

Аналіз класичних і евристичних підходів дозволяє зробити висновок, що низка важливих аспектів оптимізації логістичних маршрутів залишається невирішеною. Зокрема, існуючі методи недостатньо ефективні для багатокритеріальної оптимізації в умовах динамічних змін, характерних для реальних логістичних процесів. Також актуальною залишається проблема масштабованості алгоритмів та їх адаптації до використання у складі складних програмних систем.

Таким чином, виникає потреба у розробці або адаптації методів оптимізації, які поєднували б обчислювальну ефективність, гнучкість налаштування та можливість інтеграції в програмні логістичні платформи.

1.4. Висновки до розділу

У першому розділі магістерської роботи проведено системний аналіз предметної області оптимізації логістичних маршрутів та здійснено огляд основних наукових підходів до розв'язання задач маршрутизації. Розглянуто етапи розвитку наукової думки від класичних комбінаторних моделей до сучасних евристичних, метаевристичних та інтелектуальних методів оптимізації.

Проаналізовано базові задачі логістичної маршрутизації, зокрема задачу комівояжера та задачу маршрутизації транспортних засобів, які лежать в основі більшості практичних логістичних рішень. Досліджено класичні алгоритмічні методи та виявлено їх ключові обмеження, пов'язані з експоненційним зростанням обчислювальної складності, низькою масштабованістю та обмеженими можливостями адаптації до динамічних умов.

Критичний аналіз існуючих підходів показав, що традиційні методи оптимізації не повною мірою відповідають вимогам сучасних програмних логістичних систем, які потребують багатокритеріальної оптимізації, гнучкості, адаптивності та можливості інтеграції у складні інформаційні платформи. Це

обґрунтовує доцільність застосування метаевристичних підходів, зокрема генетичних алгоритмів, як ефективного інструмента оптимізації логістичних маршрутів.

На основі проведеного аналізу предметної області та виявлених невирішених питань у даній магістерській роботі визначено такі **основні завдання дослідження**:

- проаналізувати теоретичні основи оптимізації логістичних маршрутів;
- розглянути сучасні підходи та методи розв’язання задач маршрутизації;
- обґрунтувати доцільність використання генетичних алгоритмів для оптимізації логістичних процесів;
- побудувати формалізовану модель задачі оптимізації логістичних маршрутів.

Сформульовані завдання визначають подальший напрям дослідження та створюють методологічне підґрунтя для побудови й реалізації методу оптимізації логістичних маршрутів, що буде розглянуто у наступних розділах магістерської роботи.

РОЗДІЛ 2.

МЕТОДИ ОПТИМІЗАЦІЇ МАРШРУТІВ ІЗ ЗАСТОСУВАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ

У другому розділі розглядаються методи оптимізації логістичних маршрутів із використанням технологій штучного інтелекту. На основі аналізу класичних алгоритмів та їхніх обмежень, представлених у Розділі 1, обґрунтовано необхідність застосування AI-підходів для ефективного вирішення задач великого масштабу та динамічних умов. У цьому розділі буде детально розглянуто обраний метод — генетичний алгоритм, його принципи роботи, структура, переваги, а також порівняння з іншими AI-методами для логістики.

2.1. Огляд AI-методів у оптимізації маршрутів

Сучасні логістичні системи потребують швидкого та ефективного планування маршрутів у складних умовах із великою кількістю пунктів доставки, багатокритеріальною оптимізацією та динамічними змінами. У зв'язку з цим традиційні алгоритми часто виявляються недостатніми, що зумовлює застосування методів штучного інтелекту (AI) для оптимізації маршрутів.

Найпоширенішими AI-підходами для вирішення задач маршрутизації є генетичні алгоритми (GA), алгоритми рою частинок (PSO), імітація відпалу (Simulated Annealing), нейронні мережі та глибокі навчальні моделі. Кожен із цих методів має свої особливості та переваги.

Генетичні алгоритми засновані на принципах природного відбору та еволюції популяцій. Вони дозволяють знаходити наближені оптимальні рішення шляхом послідовних поколінь хромосом, що представляють маршрути. Основними операціями є схрещування, мутація та відбір кращих індивідумів. GA ефективні для великих задач TSP та VRP, забезпечують баланс між глобальним пошуком та

локальним покращенням, а також легко інтегруються з багатокритеріальними функціями оптимізації.

Алгоритми рою частинок (PSO) імітують поведінку груп тварин або комах, що рухаються у пошуках оптимальних рішень. Кожна частинка представляє потенційний маршрут, а рух у просторі рішень визначається її власним досвідом та досвідом сусідів. PSO дозволяє швидко знаходити наближені оптимальні маршрути, проте може застрягати у локальних мінімумах без додаткових модифікацій.

Імітація відпалу (Simulated Annealing) — це метод метаевристики, який поступово знижує ймовірність прийняття менш ефективних рішень, дозволяючи уникнути локальних мінімумів. Метод особливо ефективний для задач із великою кількістю вузлів, де необхідно покращувати початковий маршрут, отриманий іншими евристичними методами.

Нейронні мережі та глибоке навчання здатні моделювати складні залежності між вузлами та прогнозувати оптимальні маршрути на основі історичних даних. Вони добре підходять для динамічних систем, де умови часто змінюються, однак потребують значних обчислювальних ресурсів та великої кількості навчальних даних.

Порівняно з класичними методами, AI-алгоритми забезпечують ефективне масштабування для великих систем, адаптацію до динамічних змін, можливість багатокритеріальної оптимізації, скорочення часу обчислень у великих задачах через евристичні стратегії.

2.2. Генетичні алгоритми у оптимізації логістичних маршрутів

2.1.1. Генетичні алгоритми у оптимізації логістичних маршрутів

Генетичні алгоритми (GA) належать до класу евристичних та метаевристичних методів оптимізації, заснованих на принципах природного відбору, спадковості та еволюції популяцій. Вони особливо ефективні для складних NP-складних задач, таких як планування маршрутів у великих логістичних системах, де традиційні алгоритми стають непридатними через експоненційне зростання обчислювальної складності.

Основна ідея GA полягає у представленні потенційних рішень задачі у вигляді хромосом — структурованих об'єктів, що кодують маршрути. Кожна хромосома відповідає певному порядку відвідування вузлів у системі доставки. Генетичний алгоритм працює з популяцією хромосом і за допомогою еволюційних операцій поступово покращує рішення.

Основні етапи роботи GA:

1. Ініціалізація популяції. На початковому етапі створюється набір випадкових маршрутів (хромосом). Кожна хромосома є потенційним рішенням задачі оптимізації.

2. Оцінка пристосованості (fitness function). Для кожного маршруту розраховується функція пристосованості, яка відображає його якість. Для задач маршрутизації часто використовують функції, що мінімізують сумарну відстань D , час доставки T або комбіновану цільову функцію:

$$F = w_1 \cdot D + w_2 \cdot T + w_3 \cdot C \quad (2.1)$$

де F – маршрут, w_1, w_2, w_3 — вагові коефіцієнти для відстані, часу та витрат C

3. Відбір (Selection). Відбір кращих хромосом здійснюється за принципом природного відбору. Найпоширеніші методи: турнірний відбір, рулеткове коло та ранговий відбір. Мета — дати перевагу хромосомам з високою пристосованістю для подальшого схрещування.

4. Схрещування (Crossover). Цей етап імітує обмін генетичною інформацією між двома батьківськими хромосомами для створення нових маршрутів (нащадків). Існують різні методи схрещування, зокрема Order Crossover (OX), Partially Matched Crossover (PMX) та Cycle Crossover (CX). Наприклад, у PMX частина маршруту батька копіюється у нащадка, а решта вузлів переставляється так, щоб уникнути дублювання.

5. Мутація (Mutation). Мутація застосовується для збереження генетичного різноманіття та уникнення застрягання в локальних мінімумах. Найпоширеніші

операції: обмін двох вузлів, інверсія сегмента маршруту або вставка вузла у нову позицію.

6. Замінювання популяції (Replacement). Після схрещування та мутації старі хромосоми замінюються новими. Можуть застосовуватися повне заміщення або елітизм, коли найкращі хромосоми зберігаються для наступного покоління.

7. Зупинка алгоритму. Алгоритм продовжує роботу до досягнення заданого критерію зупинки: максимальна кількість поколінь, досягнення порогового значення функції пристосованості або відсутність покращення результату протягом певної кількості ітерацій.

Переваги GA для логістики очевидні. GA легко враховують динамічні зміни у маршрутах, наприклад, нові замовлення або зміни дорожньої ситуації. Працюють ефективно навіть для сотень вузлів, де класичні методи стають непридатними. Можна одночасно мінімізувати відстань, час, витрати та інші параметри. Здатність уникати локальних мінімумів, адже мутація та схрещування дозволяють досліджувати широкий простір рішень.

Розглянемо як відбувається формалізація маршруту.

Нехай є 5 пунктів доставки A, B, C, D, E . Хромосома X може бути представлена як послідовність:

$$X = [A, B, C, D, E] \quad (2.2)$$

Вона описує порядок відвідування пунктів. Функція пристосованості F може бути:

$$F = D(A \rightarrow C) + D(C \rightarrow E) + D(E \rightarrow B) + D(B \rightarrow D) + D(D \rightarrow A) \quad (2.4)$$

де $D(X \rightarrow Y)$ — відстань між пунктами X та Y .

2.1.2 Алгоритми рою частинок у оптимізації маршрутів

Алгоритми рою частинок (Particle Swarm Optimization, PSO) є одним із методів штучного інтелекту, які належать до метаевристичних підходів оптимізації, заснованих на спостереженнях за колективною поведінкою груп тварин чи комах, таких як косяки риб або зграї птахів. PSO ефективно застосовуються для вирішення складних NP-складних задач, у тому числі для оптимізації логістичних маршрутів, де класичні алгоритми та навіть деякі евристики можуть виявитися недостатньо гнучкими.

У PSO кожне потенційне рішення задачі (у нашому випадку маршрут) представлено частинкою, яка має положення та швидкість у просторі рішень. Кожна частинка запам'ятовує особистий найкращий досвід (*pbest*) – найкраще рішення, яке вона сама знайшла, а також глобальний найкращий досвід (*gbest*) – найкраще рішення серед усіх частинок рою.

Під час ітерацій частинки переміщуються у просторі рішень відповідно до формули оновлення положення та швидкості:

$$v_i(t + 1) = w \cdot v_i(t) + c_1 \cdot r_1 \cdot (pbest_i - x_i(t)) + c_2 \cdot r_2 \cdot (gbest_i - x_i(t)) \quad (2.4)$$

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (2.5)$$

де $v_i(t)$ — швидкість частинки i у поколінні t ;

- $x_i(t)$ — поточне положення частинки i ;

- w — коефіцієнт інерції, що визначає вплив попередньої швидкості;

- c_1, c_2 — коефіцієнти навчання, що регулюють вплив особистого та глобального досвіду;

- r_1, r_2 — випадкові числа у $[0,1]$, що забезпечують стохастичність руху.

Для задач маршрутизації маршрути кодуються у вигляді перестановок вузлів, а операції оновлення положення адаптуються до дискретного простору, наприклад через перестановку вузлів маршруту відповідно до напрямку руху частинки.

Етапи роботи PSO для маршрутизації

1. Ініціалізація рою. Генерується початкова популяція частинок з випадковими маршрутами.
2. Оцінка функції пристосованості. Для кожного маршруту обчислюється цільова функція, яка може включати сумарну відстань, час доставки, витрати пального або комбіновану багатокритеріальну функцію:

$$F = w_1 \cdot D + w_2 \cdot T + w_3 \cdot C \quad (2.6)$$

3. Оновлення pbest та gbest. Кожна частинка порівнює поточне рішення з власним pbest і, при необхідності, оновлює його. Також перевіряється, чи не покращилося глобальне gbest.
4. Оновлення швидкості та положення виконується за формулами оновлення з урахуванням pbest та gbest. У дискретному представленні маршруту це може бути перестановка вузлів або обмін сегментів маршруту.
5. Алгоритм працює до досягнення критерію зупинки, наприклад, максимальної кількості ітерацій або досягнення прийняттого значення функції пристосованості.

Варто підкреслити переваги застосування цього методу. PSO здатні швидко знаходити наближені оптимальні маршрути завдяки поєднанню локальної та глобальної інформації. Можуть коригувати маршрути при зміні даних у реальному часі. Простота реалізації, адже порівняно з GA, PSO має меншу кількість операторів і параметрів, що спрощує програмну реалізацію. Добре працює для середніх і великих систем доставки.

Але ці переваги нівелюються недоліками та обмеженнями методу. Це, зокрема, можливість застрягання в локальних мінімумів без додаткових модифікацій (наприклад, введення стохастичних змін у швидкості). Метод потребує точного налаштування параметрів (w, c_1, c_2) , що впливає на якість рішень. У дискретних

задачах маршрутизації потребує адаптації класичних формул PSO для перестановок вузлів.

Розглянемо роботу з маршрутом.

Нехай маємо 4 вузли доставки (A, B, C, D) . Кожна частинка може бути представлена як перестановка вузлів:

$$M = [A, C, B, D], \quad (2.7)$$

де M - маршрут частинки.

Після обчислення функції пристосованості частинка оновлює своє положення шляхом обміну вузлів відповідно до $pbest$ та $gbest$, наприклад:

$$[A, C, B, D] \rightarrow [A, B, C, D] \quad (2.8)$$

2.1.3. Нейронні мережі та глибоке навчання у оптимізації логістичних маршрутів

Нейронні мережі (Neural Networks, NN) та методи глибокого навчання (Deep Learning, DL) є сучасними підходами штучного інтелекту, які дозволяють моделювати складні нелінійні залежності між даними та приймати рішення на основі великої кількості вхідної інформації. У контексті оптимізації логістичних маршрутів вони застосовуються для прогнозування оптимальних маршрутів, класифікації транспортних потоків, оцінки часу доставки та інтеграції динамічних змін у системі.

Принцип роботи нейронних мереж. Нейронна мережа складається з шарів нейронів. Вхідний шар отримує дані про вузли маршруту, відстані, час доставки, вантажопідйомність транспортних засобів та інші параметри. Приховані шари обробляють інформацію через нелінійні функції активації (наприклад, ReLU, sigmoid, tanh), дозволяючи виявляти складні взаємозв'язки між вузлами та умовами доставки. Вихідний шар генерує прогнозоване рішення, наприклад оптимальний порядок відвідування пунктів або оцінку часу доставки.

Формально, для нейронного шару

$$y = f(\sum_{i=1}^n w_i x_i + b) \quad (2.9)$$

де x_i — вхідні дані, w_i — ваги нейрону, b — зміщення, f — функція активації, y — вихід нейрону.

Глибоке навчання та його переваги. Глибокі нейронні мережі (Deep Neural Networks, DNN) мають декілька прихованих шарів, що дозволяє ефективно виявляти складні патерни у великих наборах даних. Для задач логістики можуть застосовуватися певні типи мереж. Рекурентні нейронні мережі (RNN) та їх модифікації LSTM/GRU — для обробки послідовностей, наприклад, часових рядів замовлень чи змін у трафіку. Конволюційні нейронні мережі (CNN) — для аналізу просторових патернів, наприклад, на картографічних даних або для оцінки щільності маршрутів. Глибокі підкріплювальні мережі (Deep Reinforcement Learning, DRL) — для навчання агентів, які самостійно знаходять оптимальні маршрути на основі системи винагород, що дозволяє адаптуватися до динамічних змін.

Етапи застосування NN та DL у маршрутизації передбачають виконання наступних дій.

1. Збір та підготовка даних (історичні маршрути, відстані, час доставки, пробки, вантажі, погодні умови).
2. Формування вхідних та вихідних даних (кодування вузлів маршруту, нормалізація числових показників, створення послідовностей для RNN).
3. Навчання моделі (оптимізація ваг мережі за допомогою алгоритму зворотного поширення помилки (backpropagation) та функції втрат, наприклад, середньоквадратичної помилки або спеціальної багатокритеріальної функції для маршрутизації).
4. Тестування та валідація (оцінка точності прогнозів на нових маршрутах, перевірка адаптивності до змінних умов).

5. Використання для планування маршрутів (мережа генерує оптимальні або наближені маршрути у реальному часі, враховуючи динамічні зміни та багатокритеріальні умови).

Переваги нейронних мереж та DL для логістики - обробка великих даних (здатність працювати з великими обсягами історичних та поточних даних); динамічність (можливість швидко реагувати на зміни у системі доставки); прогнозування (точна оцінка часу доставки, пробок, навантаження транспортних засобів); багатокритеріальна оптимізація (інтеграція декількох параметрів у функцію втрат); навчання на досвіді (здатність підвищувати якість маршрутів на основі історичних даних та винагород).

Проте робота з цим методом має ряд недоліків, які слід враховувати. Одним із найвагоміших є високі обчислювальні ресурси, адже для тренування глибоких мереж потрібні GPU та великі обсяги пам'яті. Варто врахувати необхідність великої кількості даних, оскільки ефективність моделі зростає із збільшенням обсягу навчальної інформації. Ще одним недоліком є складність інтерпретації – рішення нейронної мережі важко пояснюються порівняно з традиційними методами. Але найбільшим недоліком є те, що вибір кількості шарів, нейронів, швидкості навчання та функцій активації критично впливає на результати.

Для обґрунтування доцільності використання різних AI-методів у задачах оптимізації логістичних маршрутів доцільно провести їх порівняльний аналіз. Кожен із підходів має власні переваги та обмеження, що визначають ефективність його застосування залежно від масштабу задачі, динамічності середовища та кількості критеріїв оптимізації. Узагальнені характеристики основних AI-методів подано в таблиці 2.1.

Аналіз даних, наведених у таблиці 2.1, показує, що жоден із розглянутих AI-методів не є універсальним рішенням для всіх задач оптимізації логістичних маршрутів. Генетичні алгоритми характеризуються високою гнучкістю, масштабованістю та здатністю уникати локальних мінімумів, що робить їх ефективними для задач великої розмірності та багатокритеріальної оптимізації.

Водночас вони потребують ретельного налаштування параметрів та можуть мати повільнішу збіжність у порівнянні з деякими іншими методами

Таблиця 2.1

Порівняння AI-методів для оптимізації логістичних маршрутів на основі описаних підрозділів

Метод AI	Переваги	Недоліки
Генетичні алгоритми (GA)	Гнучкість, масштабованість, здатність уникати локальних мінімумів, багатокритеріальна оптимізація	Можлива повільна збіжність на великих задачах, потреба налаштування параметрів (розмір популяції, ймовірності мутації/схрещування)
Алгоритми рою частинок (PSO)	Швидка збіжність, простота реалізації, адаптивність до змін, масштабованість	Можливість застрягання в локальних мінімумів, потреба тонкого налаштування параметрів (w , $c1$, $c2$), складність у дискретному представленні маршруту
Нейронні мережі та глибоке навчання (NN/DL)	Обробка великих даних, динамічність, точне прогнозування, багатокритеріальна оптимізація, навчання на досвіді	Високі обчислювальні ресурси, потреба великої кількості даних, складність інтерпретації, необхідність налаштування гіперпараметрів

Алгоритми рою частинок та імітації відпалу демонструють швидшу збіжність і простішу реалізацію, проте їх ефективність знижується у випадках складних дискретних обмежень і високої багатокритеріальності. Нейронні мережі та методи глибокого навчання забезпечують високу адаптивність і здатність працювати з динамічними даними, однак потребують значних обчислювальних ресурсів і великих навчальних вибірок, що обмежує їх застосування у практичних логістичних системах без відповідної інфраструктури. Таким чином, результати порівняння підтверджують доцільність вибору генетичного алгоритму як базового методу оптимізації

логістичних маршрутів у межах даної магістерської роботи. Більш детально вибір буде обґрунтовано в наступному підрозділі

2.3. Обґрунтування вибору генетичного алгоритму

Вибір конкретного методу оптимізації логістичних маршрутів у магістерській роботі ґрунтується на аналізі особливостей задачі, обмежень класичних методів та порівнянні сучасних AI-технологій. Як було показано у підрозділі 2.1, серед популярних AI-підходів для маршрутизації виділяють генетичні алгоритми (GA), алгоритми рою частинок (PSO) та нейронні мережі/глибоке навчання (NN/DL). Кожен із методів має свої переваги та обмеження, що визначає доцільність їх використання залежно від специфіки логістичної системи.

Генетичні алгоритми були обрані для практичної реалізації з кількох причин. По-перше, вони поєднують високу гнучкість із ефективним пошуком наближених оптимальних рішень. GA дозволяють працювати з великими системами доставки, включно з сотнями вузлів, завдяки використанню популяційного підходу та евристичних операцій схрещування і мутації, які зменшують ризик застрягання в локальних мінімумах.

По-друге, GA дозволяють інтегрувати багатокритеріальні цільові функції, що є важливим у сучасних логістичних задачах. Наприклад, можна одночасно мінімізувати сумарну відстань, час доставки та витрати на паливо, а також враховувати часові вікна для обслуговування клієнтів. Така здатність до багатокритеріальної оптимізації є перевагою перед PSO, яка може потребувати додаткових модифікацій для ефективного балансування кількох показників.

По-третє, генетичні алгоритми простіші в програмній реалізації та налагодженні порівняно з нейронними мережами та глибоким навчанням. NN/DL вимагають великої кількості даних для навчання, значних обчислювальних ресурсів та складного підбору гіперпараметрів, що може ускладнювати швидку реалізацію прототипу в умовах магістерської роботи. GA ж забезпечують швидку інтеграцію з

існуючими моделями даних та достатньо прості для експериментальної перевірки ефективності.

Крім того, GA легко адаптуються до специфічних обмежень логістичних задач, таких як вантажопідйомність транспортних засобів, обмеження на час доставки або необхідність об'їзду певних маршрутів. Шляхом введення штрафних функцій у цільову функцію або модифікації операторів мутації та схрещування можна швидко врахувати ці обмеження без зміни основної структури алгоритму.

Обґрунтування вибору генетичного алгоритму для оптимізації логістичних маршрутів ґрунтується на п'яти ключових факторах, які забезпечують ефективність та практичну доцільність його використання:

1. Ефективність для великих задач та здатність знаходити наближені оптимальні рішення у розумні терміни;
2. Можливість багатокритеріальної оптимізації з урахуванням відстані, часу та витрат;
3. Простота реалізації та налагодження порівняно з нейронними мережами та PSO;
4. Гнучкість щодо специфічних обмежень логістичних систем;
5. Масштабованість та адаптивність до динамічних змін у маршрутах.

Розглянемо детальніше кожен із них.

Генетичні алгоритми працюють з популяцією потенційних рішень, що дозволяє одночасно досліджувати великий простір можливих маршрутів. Класичні методи, такі як повний перебір або динамічне програмування, швидко стають непридатними при збільшенні кількості вузлів через експоненційне зростання складності задачі (NP-складність).

GA дозволяють знаходити наближені оптимальні рішення у розумний проміжок часу, навіть для задач із сотнями пунктів доставки. Кожне покоління алгоритму відбирає кращі рішення та комбінує їх для створення нових маршрутів, що поступово покращує якість рішень без необхідності повного перебору всіх варіантів.

Логістичні задачі зазвичай мають декілька критеріїв оптимізації: сумарну відстань, час доставки, витрати на паливо, екологічні показники та дотримання часових вікон. Генетичні алгоритми дозволяють формалізувати цільову функцію як комбінацію цих критеріїв:

$$F = w_1 \cdot D + w_2 \cdot T + w_3 \cdot C + w_4 \cdot P, \quad (2.9)$$

де

- D — сумарна відстань;
- T — час доставки;
- C — витрати на паливо;
- P — штрафи за порушення обмежень;
- w_i — вагові коефіцієнти.

Така гнучкість дозволяє моделювати різні пріоритети для різних задач та клієнтів.

GA мають інтуїтивно зрозумілу структуру: хромосоми кодують маршрути, операції схрещування і мутації легко реалізуються програмно, а функція пристосованості оцінює якість маршруту. Порівняно з нейронними мережами та методами глибокого навчання, GA не потребують великих обсягів даних для навчання і складного налаштування гіперпараметрів, що дозволяє швидко створити прототип та перевірити його ефективність.

Реальні логістичні системи мають численні обмеження: вантажопідйомність транспортних засобів; часові вікна для доставки; об'їзд заборонених зон або доріг; пріоритети клієнтів.

GA дозволяють вводити ці обмеження через модифікацію функції пристосованості (штрафні функції) або адаптацію операторів мутації та схрещування. Наприклад, маршрути, які порушують часові вікна, отримують штраф, що знижує їхню ймовірність відбору.

Генетичні алгоритми ефективно масштабуються від невеликих систем (до 20–30 вузлів) до великих систем із сотнями вузлів. Крім того, вони добре адаптуються до динамічних змін: нові замовлення або зміни дорожньої ситуації можна інтегрувати без повного перерахунку всіх маршрутів. Достатньо оновити популяцію хромосом і продовжити еволюційний процес.

Генетичний алгоритм (GA) є ефективним методом оптимізації, який імітує природні процеси еволюції та відбору. Основна ідея полягає у тому, що потенційні рішення задачі представляються у вигляді хромосом — структурованих об'єктів, що кодуєть маршрути. Кожна хромосома містить впорядковану послідовність вузлів доставки, а популяція хромосом одночасно досліджує великий простір можливих рішень. Такий підхід дозволяє знаходити наближені оптимальні маршрути навіть для великих систем, де класичні методи перебору стають непридатними через експоненційне зростання складності.

Реалізація GA базується на циклічному процесі еволюції популяції. На початковому етапі формується набір випадкових маршрутів, кожен з яких оцінюється за функцією пристосованості. Ця функція може включати сумарну відстань, час доставки, витрати та інші обмеження, наприклад, вантажопідйомність транспортних засобів або часові вікна. Використання багатокритеріальної функції дозволяє алгоритму одночасно оптимізувати декілька параметрів, забезпечуючи більш ефективні та практично значущі рішення.

Ключовими операторами GA є відбір, схрещування та мутація. Відбір визначає, які хромосоми будуть брати участь у наступному поколінні, надаючи перевагу більш пристосованим рішенням. Схрещування поєднує інформацію від двох батьківських маршрутів, формуючи нові хромосоми-нащадки, тоді як мутація випадковим чином змінює порядок вузлів у маршруті, підтримуючи різноманітність популяції та запобігаючи застряганню в локальних мінімумів. Існують різні методи схрещування та мутації, адаптовані для дискретних задач маршрутизації, наприклад, Order Crossover (OX), Partially Matched Crossover (PMX) або інверсія сегмента маршруту.

Процес еволюції повторюється протягом багатьох поколінь, поки не буде досягнуто критерію зупинки, який може базуватися на максимальній кількості ітерацій, досягненні порогового значення функції пристосованості або відсутності покращень протягом кількох поколінь. На виході алгоритм надає найкращу хромосому, яка представляє оптимальний або наближений оптимальний маршрут доставки.

Особливістю генетичного алгоритму є його гнучкість та адаптивність. ГА можна легко масштабувати для роботи з великими системами, інтегрувати багатокритеріальні функції, враховувати специфічні обмеження логістики, а також адаптувати до динамічних змін у маршрутах. Завдяки поєднанню популяційного пошуку, еволюційних операторів та багатокритеріальної оцінки рішень, генетичний алгоритм забезпечує ефективну оптимізацію логістичних маршрутів і є доцільним для реалізації у практичній частині магістерської роботи.

2.4. Архітектурна організація та логіка поетапної реалізації генетичного алгоритму

Генетичний алгоритм у контексті інженерії програмного забезпечення доцільно розглядати не лише як математичний або евристичний метод оптимізації, а як програмну систему з чітко визначеною внутрішньою структурою, логікою взаємодії компонентів та послідовністю виконання етапів. Такий підхід дозволяє узагальнити процес його реалізації, абстрагуватися від конкретної предметної області та забезпечити можливість повторного використання алгоритму в різних прикладних задачах.

З архітектурної точки зору генетичний алгоритм являє собою ітеративну систему, у якій кожен етап виконує окрему функціональну роль, а результат роботи попереднього етапу є входними даними для наступного. Сукупність цих етапів формує еволюційний цикл, що повторюється до моменту досягнення заданих умов завершення. Важливою особливістю такої організації є її модульність, що відповідає принципам сучасного програмного проектування.

Початковим елементом архітектури генетичного алгоритму є модуль ініціалізації, який відповідає за формування початкової популяції можливих рішень. На цьому етапі визначається спосіб кодування рішень у вигляді хромосом, а також правила формування окремих генів. У загальному випадку ініціалізація здійснюється випадковим чином, що дозволяє забезпечити різноманіття початкових рішень та уникнути упередженості пошуку. З точки зору програмної реалізації, даний етап задає початковий стан системи та визначає межі простору пошуку.

Наступним важливим компонентом є модуль оцінювання, який реалізує функцію пристосованості. Саме на цьому етапі здійснюється зв'язок між абстрактним представленням рішень і конкретними вимогами задачі оптимізації. Функція пристосованості перетворює кожну хромосому на числову оцінку, що характеризує її якість. В архітектурному сенсі цей модуль є центральним, оскільки від коректності його реалізації залежить ефективність усього алгоритму. Крім того, він часто є найбільш обчислювально затратним, що вимагає особливої уваги при програмній реалізації.

На основі результатів оцінювання функціонує модуль відбору, який реалізує механізм еволюційного тиску. Його призначення полягає у визначенні тих особин, які братимуть участь у формуванні нового покоління. Важливо підкреслити, що відбір не є жорстким відсівом усіх слабких рішень, а передбачає ймовірнісний характер вибору. Це дозволяє зберігати генетичне різноманіття популяції та запобігати передчасній збіжності алгоритму. В контексті програмної інженерії модуль відбору реалізує логіку прийняття рішень на основі оцінок якості.

Після відбору активується модуль генетичних операторів, який включає оператори схрещування та мутації. Схрещування забезпечує комбінування генетичної інформації двох або більше батьківських рішень, створюючи нові комбінації параметрів. З програмної точки зору це означає реалізацію механізмів обміну фрагментами структур даних. Мутація, у свою чергу, відповідає за внесення випадкових змін до окремих генів, що сприяє дослідженню нових областей простору

рішень. Незважаючи на допоміжний характер, мутація відіграє критичну роль у підтриманні ефективності алгоритму.

Результатом роботи генетичних операторів є формування кандидатів нового покоління, після чого відбувається оновлення популяції. Цей етап визначає, які саме особини переходять до наступної ітерації еволюційного циклу. Залежно від обраної стратегії, оновлення може включати збереження найкращих рішень попереднього покоління, що дозволяє поєднати стабільність і прогрес у процесі оптимізації. В архітектурному плані це відповідає управлінню життєвим циклом об'єктів у системі.

Завершальним елементом є модуль контролю умов зупинки, який аналізує поточний стан алгоритму та приймає рішення про завершення або продовження еволюційного процесу. До умов завершення можуть належати досягнення заданої кількості ітерацій, стабілізація значень функції пристосованості або виконання наперед визначених критеріїв якості. Реалізація цього модуля забезпечує керованість алгоритму та передбачуваність його роботи.

Узагальнюючи, архітектурна організація генетичного алгоритму являє собою послідовну та логічно впорядковану систему взаємодіючих модулів, кожен з яких виконує чітко визначену функцію. Такий підхід дозволяє розглядати генетичний алгоритм не лише як теоретичний метод оптимізації, а як основу для побудови програмних рішень, орієнтованих на складні багатокритеріальні задачі. Саме ця особливість створює передумови для практичного застосування генетичного алгоритму в задачах оптимізації логістичних маршрутів, що буде послідовно розкрито в наступному розділі роботи.

2.5. Порівняння генетичного алгоритму з іншими AI-методами

Для обґрунтування вибору генетичного алгоритму (GA) у практичній частині магістерської роботи доцільно порівняти його з іншими популярними методами штучного інтелекту, які застосовуються для оптимізації логістичних маршрутів: алгоритмами рою частинок (PSO) та нейронними мережами/глибоким навчанням (NN/DL). Порівняння здійснюється за кількома ключовими параметрами:

ефективність пошуку, здатність до багатокритеріальної оптимізації, ресурсоміткість, адаптивність до змін та складність реалізації.

1. Ефективність пошуку та збіжність

Генетичні алгоритми ефективно досліджують великий простір рішень завдяки популяційному підходу та еволюційним операціям. GA забезпечують швидкий наближений пошук оптимальних маршрутів, особливо для задач середнього та великого масштабу. PSO зазвичай демонструє швидку збіжність, оскільки частинки коригують свої маршрути на основі локального та глобального досвіду, проте у великих дискретних просторах може застрягати в локальних мінімумів без спеціальних модифікацій. Нейронні мережі та глибоке навчання забезпечують високу точність прогнозів і можуть оптимізувати маршрути в реальному часі, проте ефективність залежить від обсягу даних і складності моделі, що робить їх більш ресурсоміткими.

2. Багатокритеріальна оптимізація

GA відзначаються здатністю інтегрувати кілька критеріїв у функцію пристосованості, наприклад сумарну відстань, час доставки, витрати та штрафи за порушення обмежень. PSO також може реалізовувати багатокритеріальну оптимізацію, але потребує додаткових модифікацій для балансування критеріїв. Нейронні мережі можуть працювати з багатокритеріальними функціями через відповідні цільові функції або підкріплювальне навчання, але потребують значного обсягу даних і налаштування гіперпараметрів.

3. Простота реалізації та ресурсоміткість

GA є порівняно простими в реалізації: операції схрещування, мутації та відбору легко програмуються, не потребують навчання на великих масивах даних та спеціального апаратного забезпечення. PSO теж має просту реалізацію, але у дискретних задачах маршрутизації його формули потребують адаптації. Нейронні мережі та глибоке навчання вимагають потужних обчислювальних ресурсів, спеціального програмного забезпечення, та складного налаштування архітектури і гіперпараметрів.

4. Адаптивність та масштабованість

GA та PSO добре масштабуються для великих систем доставки і можуть адаптуватися до змін у маршрутах, наприклад, при появі нових замовлень або зміні дорожньої ситуації. Нейронні мережі також можуть враховувати динамічні зміни, але потребують постійного донавчання на актуальних даних, що збільшує обчислювальне навантаження. GA дозволяють швидко інтегрувати обмеження логістики, наприклад вантажопідйомність або часові вікна, шляхом модифікації функції пристосованості або операторів мутації.

5. Узагальнена оцінка

Порівняльний аналіз показує, що генетичний алгоритм оптимально поєднує гнучкість, простоту реалізації, ефективність пошуку та багатокритеріальну оптимізацію для задач маршрутизації. PSO підходить для швидкого наближеного пошуку в середніх системах, але потребує додаткових адаптацій у дискретних умовах. NN/DL ефективні для складних прогностичних задач і динамічних систем з великими даними, але є ресурсомісткими і складними у реалізації для прототипу.

Генетичний алгоритм демонструє найкращий баланс між ефективністю, гнучкістю та простотою реалізації для конкретної магістерської роботи. Він дозволяє інтегрувати багатокритеріальні функції, враховувати логістичні обмеження, швидко адаптуватися до змін та забезпечує наочну експериментальну перевірку у практичній частині. Тому GA обраний для подальшої експериментальної реалізації системи оптимізації маршрутів у Розділі 3.

2.6 Висновки до розділу

У другому розділі магістерської роботи було розглянуто алгоритмічні та програмні аспекти оптимізації логістичних маршрутів із використанням методів штучного інтелекту. Проаналізовано особливості задач маршрутизації як об'єкта програмної реалізації, визначено її основні обмеження та критерії оптимізації, що є необхідним для побудови ефективних алгоритмічних рішень.

Обґрунтовано доцільність застосування генетичного алгоритму для розв'язання задач оптимізації логістичних маршрутів у програмних системах, зважаючи на його гнучкість, масштабованість та здатність працювати з багатокритеріальними задачами. Розглянуто принципи побудови генетичного алгоритму та визначено ключові етапи його роботи, які можуть бути реалізовані у вигляді окремих програмних модулів.

У розділі сформовано формалізовану модель задачі оптимізації логістичних маршрутів, що включає опис вхідних даних, обмежень та критеріїв оцінювання якості рішень. Отримані результати створюють теоретичне та алгоритмічне підґрунтя для подальшої програмної реалізації та проведення експериментальних досліджень ефективності запропонованого підходу, які розглядаються у наступних розділах роботи.

РОЗДІЛ 3

ПОБУДОВА МЕТОДУ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ ІЗ ЗАСТОСУВАННЯМ ГЕНЕТИЧНОГО АЛГОРИТМУ

У третьому розділі магістерської роботи розглядається побудова методу оптимізації логістичних маршрутів із використанням генетичного алгоритму як базового інструмента реалізації. Основна увага зосереджується на методологічних аспектах оптимізації, зокрема на формуванні критеріїв ефективності маршрутів, побудові цільової функції та алгоритмічній логіці пошуку оптимальних рішень, а не на деталях програмної реалізації.

Запропонований метод орієнтований на розв'язання задач маршрутизації в умовах багатокритеріальності та обмежень, характерних для сучасних логістичних систем. Генетичний алгоритм використовується як універсальний еволюційний підхід, що дозволяє реалізувати процес оптимізації шляхом послідовного покращення рішень без необхідності повного перебору можливих маршрутів.

У межах розділу послідовно розглядаються концепція методу оптимізації маршрутів, критерії та цільова функція, алгоритмічна схема роботи методу, а також його переваги й обмеження. Такий підхід дозволяє продемонструвати логіку побудови методу оптимізації та обґрунтувати доцільність застосування генетичного алгоритму для розв'язання задач оптимізації логістичних маршрутів у контексті інженерії програмного забезпечення.

3.1. Загальна концепція методу оптимізації логістичних маршрутів

Метод оптимізації логістичних маршрутів у межах даної магістерської роботи розглядається як формалізований підхід до побудови ефективних маршрутів доставки, що базується на поєднанні багатокритеріального аналізу та еволюційних принципів пошуку рішень. Основною метою методу є визначення такого маршруту,

який забезпечує оптимальне співвідношення між витратами ресурсів, часовими показниками та експлуатаційними обмеженнями логістичної системи.

З методологічної точки зору задача оптимізації маршрутів може бути подана як задача пошуку оптимального рішення в просторі допустимих комбінацій. Кожне рішення відповідає певному маршруту, який визначається послідовністю пунктів доставки та характеристиками переміщення між ними. У реальних логістичних умовах кількість таких комбінацій є надзвичайно великою, що зумовлює необхідність застосування методів наближеного пошуку, здатних працювати з великими просторами рішень без повного перебору.

Запропонований метод ґрунтується на еволюційному підході, відповідно до якого оптимальне рішення формується поступово шляхом ітеративного покращення множини можливих маршрутів. У межах методу всі маршрути розглядаються як кандидати на оптимальне рішення та підлягають оцінюванню за визначеними критеріями ефективності. Такий підхід дозволяє абстрагуватися від конкретної реалізації та зосередитися на загальній логіці пошуку оптимальних маршрутів.

Генетичний алгоритм у структурі методу виступає як інструмент реалізації еволюційного пошуку, що забезпечує формалізацію процесу оптимізації у вигляді послідовних етапів: формування початкової множини маршрутів, оцінювання їх якості, відбір перспективних рішень та генерація нових варіантів маршрутів. При цьому сам метод не обмежується конкретними параметрами алгоритму, що забезпечує його адаптивність та можливість модифікації відповідно до особливостей конкретної логістичної задачі.

Особливу увагу в методі приділено багатокритеріальності оптимізації. На відміну від підходів, що орієнтовані виключно на мінімізацію довжини маршруту або часу доставки, запропонований метод дозволяє одночасно враховувати декілька факторів ефективності, які можуть мати різну пріоритетність залежно від умов експлуатації. Це забезпечує більшу гнучкість методу та підвищує практичну значущість отриманих результатів.

З позицій інженерії програмного забезпечення метод оптимізації розглядається як окремий логічний компонент програмної системи, який може бути інтегрований у складі інформаційних логістичних платформ. Чітке розмежування між концепцією методу та алгоритмічною реалізацією дозволяє забезпечити масштабованість, повторне використання та розширюваність програмних рішень, побудованих на його основі.

Таким чином, загальна концепція методу оптимізації логістичних маршрутів полягає у поєднанні еволюційного підходу до пошуку рішень з багатокритеріальним оцінюванням маршрутів, де генетичний алгоритм виступає ефективним механізмом реалізації методу. Такий підхід створює методологічне підґрунтя для подальшого формалізованого опису критеріїв оптимізації та побудови алгоритму реалізації методу.

3.2 Критерії оптимізації та їх формалізація

Оптимізація логістичних маршрутів у межах запропонованого методу ґрунтується на використанні сукупності критеріїв, які відображають ефективність маршруту з точки зору витрат ресурсів, часу виконання та експлуатаційних характеристик логістичної системи. Формалізація таких критеріїв є необхідною умовою застосування еволюційних методів оптимізації, оскільки саме на їх основі здійснюється оцінювання якості кожного можливого маршруту.

У загальному випадку маршрут може бути поданий як впорядкована послідовність пунктів доставки

$$R = \{v_0, v_1, v_2, \dots, v_n\} \quad (3.1)$$

де v_0 — початковий пункт (депо), а v_i — окремі точки доставки. Для кожного маршруту визначається значення цільової функції, яка інтегрує обрані критерії оптимізації у єдиний показник якості.

Одним із базових критеріїв є мінімізація загальної довжини маршруту. Цей критерій відображає сукупну відстань, яку необхідно подолати транспортному засобу, та безпосередньо впливає на витрати пального і зношування транспорту. Формально довжина маршруту може бути визначена як сума відстаней між послідовними пунктами:

$$D(R) = \sum_{i=0}^{n-1} d(v_i, v_{i+1}), \quad (3.2)$$

де $d(v_i, v_{i+1})$ — відстань між пунктами маршруту.

Другим важливим критерієм є мінімізація часу доставки, який включає не лише час руху між пунктами, а й можливі затримки, пов'язані з особливостями дорожньої інфраструктури або обслуговуванням клієнтів. Час виконання маршруту може бути поданий у вигляді:

$$T(R) = \sum_{i=0}^{n-1} t(v_i, v_{i+1}) + \sum_{i=1}^n t_s(v_i), \quad (3.3)$$

де $t(v_i, v_{i+1})$ — час переміщення між пунктами, а $t_s(v_i)$ — час обслуговування у відповідній точці.

Ще одним критерієм, який враховується у запропонованому методі, є економічна ефективність маршруту, що може бути виражена через загальні витрати на транспортування. До таких витрат можуть входити витрати на паливе, оплату праці водія, експлуатаційні витрати та штрафи за порушення часових обмежень. Узагальнено цей критерій можна представити у вигляді функції:

$$C(R) = f(D(R), T(R), \text{інші витрати}), \quad (3.4)$$

що дозволяє інтегрувати різні складові у єдиний показник.

Оскільки задача оптимізації маршрутів має багатокритеріальний характер, у межах методу застосовується зважена цільова функція (3.1). Значення вагових

коефіцієнтів визначаються залежно від специфіки логістичної задачі та можуть змінюватися без модифікації структури самого методу.

Отримане значення цільової функції використовується як функція пристосованості у генетичному алгоритмі, що дозволяє порівнювати маршрути між собою та здійснювати відбір найбільш ефективних рішень. Чим меншим є значення функції $F(R)$, тим вищою вважається якість відповідного маршруту.

З позицій інженерії програмного забезпечення формалізація критеріїв оптимізації забезпечує чіткий інтерфейс між логікою методу та алгоритмічною реалізацією. Такий підхід дозволяє легко розширювати або змінювати набір критеріїв без порушення загальної структури алгоритму, що підвищує гнучкість та адаптивність програмної реалізації методу.

Таким чином, формалізоване подання критеріїв оптимізації створює математичну основу для застосування генетичного алгоритму та забезпечує коректне оцінювання маршрутів у процесі еволюційного пошуку оптимального рішення.

3.3 Реалізація критеріїв оптимізації у структурі генетичного алгоритму

У межах запропонованого методу оптимізації логістичних маршрутів кожен можливий маршрут розглядається як окреме рішення, яке можна формалізувати у вигляді хромосоми для генетичного алгоритму. Такий підхід дозволяє алгоритму ефективно оперувати маршрутами, оцінювати їх якість та поступово знаходити оптимальні рішення без необхідності повного перебору всіх можливих комбінацій.

Перший критерій – це представлення маршрутів. Кожен маршрут кодується у вигляді впорядкованої послідовності пунктів доставки, починаючи з депо та завершуючи останньою точкою маршруту. У циклічних маршрутах транспортний засіб повертається до початкового пункту. Хромосома в такому представленні відображає весь шлях, який проходить транспортний засіб, що дозволяє алгоритму працювати з маршрутами як з цілісними рішеннями.

Для визначення якості маршруту використовуються кілька ключових критеріїв, які дозволяють оцінити його ефективність з різних точок зору. Кожен

критерій вносить свій вклад у загальну функцію оцінки маршруту, що формує основу для роботи генетичного алгоритму.

Цей критерій відображає, яку відстань проходить транспортний засіб під час доставки. Чим коротший маршрут, тим менше витрати пального, зношування транспорту та витрати часу на переміщення. Генетичний алгоритм оцінює кожен маршрут за цим показником, надаючи пріоритет коротшим маршрутам при відборі для подальшого схрещування та мутації. Мінімізація довжини маршруту є інтуїтивним і практично застосовним критерієм, який легко інтегрується з іншими показниками ефективності.

Час доставки враховує, за який проміжок часу транспортний засіб проходить всі пункти маршруту і повертається до початкового пункту. Він впливає на рівень сервісу, дозволяє виконувати замовлення вчасно та враховувати часові обмеження для клієнтів. У генетичному алгоритмі маршрути з меншим часом доставки отримують вищу оцінку, що підвищує ймовірність їх відбору і створення нових оптимальних рішень. Цей критерій забезпечує баланс між швидкістю виконання маршрутів та іншими показниками ефективності.

Додатковим критерієм є економічна ефективність маршруту, яка включає витрати на паливо, оплату праці водія, експлуатаційні витрати та інші ресурси. Врахування цього критерію дозволяє вибирати маршрути, які не лише короткі та швидкі, але й економічно доцільні. Інтеграція економічних витрат у функцію оцінки робить метод більш наближеним до реальних умов логістики.

Відбір маршрутів для наступного покоління здійснюється на основі оцінки функції пристосованості, яка об'єднує всі критерії. Кращі маршрути мають більшу ймовірність бути обраними для схрещування та мутації. Схрещування дозволяє комбінувати ділянки різних маршрутів, а мутація забезпечує випадкові зміни для уникнення локальних мінімумів. Так алгоритм поступово знаходить більш ефективні маршрути.

Багатокритеріальність реалізується через вагові коефіцієнти, які визначають пріоритетність кожного критерію. Це дозволяє адаптувати метод до різних умов

логістичної системи та вимог замовника. При цьому структура алгоритму залишається незмінною, що забезпечує його гнучкість і масштабованість.

Таблиця 3.1.

Вербальна схема роботи генетичного алгоритму для оптимізації маршрутів

Етап алгоритму	Опис дій
Формування початкової популяції	Генерується набір маршрутів, кожен з яких кодується як хромосома, що представляє послідовність пунктів доставки.
Оцінка маршрутів	Для кожного маршруту обчислюються критерії ефективності: загальна довжина маршруту, час доставки, економічні витрати. Результати інтегруються у багатокритеріальну функцію оцінки.
Відбір кращих маршрутів	На основі оцінки функції пристосованості обираються маршрути, які будуть батьківськими для наступного покоління. Кращі маршрути мають більшу ймовірність брати участь у схрещуванні.
Схрещування	Комбінуються частини маршрутів різних хромосом для створення нових рішень, поєднуючи оптимальні ділянки маршрутів.
Мутація	Випадковим чином змінюються елементи маршруту, щоб уникнути локальних мінімумів і розширити простір пошуку рішень.
Формування нового покоління	Нові маршрути замінюють частину попередньої популяції, після чого оцінюються за тією ж функцією пристосованості.
Перевірка критерію завершення	Якщо досягнуто максимальну кількість поколінь або покращення маршрутів зупинилося, алгоритм завершує роботу; інакше повертається до оцінки маршрутів для нового покоління.
Отримання оптимального маршруту	Обирається маршрут з найвищою оцінкою функції пристосованості, що є оптимальним за сукупністю визначених критеріїв.

Робота алгоритму триває до досягнення одного з критеріїв завершення: максимальна кількість поколінь, отримання маршруту з прийнятною оцінкою або відсутність істотних покращень у останніх поколіннях. На завершальному етапі

алгоритм повертає маршрут, який є найефективнішим за сукупністю обраних критеріїв.

Як бачимо, генетичний алгоритм працює як послідовність логічних кроків, де кожна операція — від формування популяції до вибору оптимального маршруту — має чітку роль у процесі еволюційного пошуку.

Таблиця 3.1 демонструє, що робота генетичного алгоритму починається з формування початкової популяції маршрутів, кожен з яких кодується як хромосома. Далі для кожного маршруту обчислюються критерії ефективності — довжина маршруту, час доставки та економічні витрати. На основі цих оцінок відбираються кращі маршрути, які потім беруть участь у схрещуванні та мутації для створення нового покоління рішень. Процес повторюється до досягнення заданого критерію завершення, після чого обирається оптимальний маршрут.

3.4 Послідовність роботи алгоритму оптимізації маршрутів

Після опису структури генетичного алгоритму та критеріїв оцінки маршрутів важливо зрозуміти, як алгоритм працює на практиці, тобто як реалізується послідовність дій для знаходження оптимального маршруту. Нами представлено логічну послідовність роботи алгоритму (табл. 3.2.), інтегровану із застосуванням описаних критеріїв: мінімізації довжини маршруту, часу доставки та економічних витрат.

На першому етапі визначаються всі вихідні дані для побудови маршрутів: перелік пунктів доставки, включно з початковим пунктом (депо); відстані між усіма пунктами; очікуваний час обслуговування кожного пункту; доступні транспортні ресурси; обмеження, наприклад, максимальний час маршруту або пріоритетні точки доставки.

Цей етап є критичним, оскільки точність та повнота даних безпосередньо впливають на якість кінцевого рішення.

На другому етапі формується початкова популяція маршрутів. Кожен маршрут кодується як хромосома — впорядкована послідовність пунктів доставки. Популяція

повинна бути різноманітною, щоб охопити широкий простір рішень і забезпечити ефективний пошук оптимальних маршрутів.

Таблиця 3.2

Послідовність роботи генетичного алгоритму оптимізації маршрутів

Етап алгоритму	Опис дій
1. Підготовка даних	Визначаються пункти доставки, відстані між ними, час обслуговування, наявні ресурси та обмеження маршруту.
2. Ініціалізація популяції	Формується початкова популяція маршрутів, кожен маршрут кодується як хромосома.
3. Оцінка маршрутів	Для кожного маршруту обчислюються критерії ефективності: довжина, час доставки, економічні витрати; формується багатокритеріальна оцінка.
4. Відбір кращих маршрутів	Вибираються маршрути для схрещування на основі їх функції пристосованості; кращі рішення мають більшу ймовірність відбору.
5. Схрещування	Комбінуються частини маршрутів різних хромосом для формування нових маршрутів із потенційно кращими характеристиками.
6. Мутація	Випадковим чином змінюються елементи маршруту, щоб уникнути локальних мінімумів та збільшити різноманітність рішень.
7. Формування нового покоління	Нові маршрути замінюють частину попередньої популяції; повторюється оцінка та відбір для наступного покоління.
8. Критерій завершення	Алгоритм зупиняється при досягненні максимального числа поколінь, стабілізації показників або отриманні прийняттого маршруту.
9. Вибір оптимального маршруту	Обирається маршрут із найвищою багатокритеріальною оцінкою, який вважається оптимальним за усіма критеріями.

Для кожної хромосоми обчислюється багатокритеріальна оцінка: довжина маршруту; час доставки; економічні витрати. Оцінка дозволяє порівняти маршрути між собою і визначити, які з них є більш ефективними для подальшого еволюційного процесу.

Далі на основі оцінки функції пристосованості обираються маршрути для схрещування. Кращі маршрути мають вищу ймовірність потрапити у наступне покоління. Відбір забезпечує збереження ефективних рішень і спрямовує алгоритм у бік оптимізації.

Наступний етап – схрещення та мутація. Схрещування дозволяє поєднувати частини різних маршрутів для створення нових комбінацій, що потенційно зменшують довжину та час доставки.

Мутація вносить випадкові зміни у маршрути для запобігання застою в локальних мінімумах і підтримує різноманітність популяції. Ці оператори забезпечують еволюційне покращення маршрутів та поступове наближення до оптимального рішення.

Наступний етап передбачає формування нового покоління. Нові маршрути замінюють частину попередньої популяції. Далі алгоритм повторно оцінює всі маршрути за обраними критеріями, відбирає кращі та застосовує еволюційні оператори. Цикл повторюється до досягнення критерію завершення.

Алгоритм припиняє роботу, коли досягається один з критеріїв завершення: максимальна кількість поколінь; стабілізація значень функції пристосованості; отримання маршруту з прийнятним рівнем оцінки за всіма критеріями.

На завершальному етапі обирається маршрут, який має найкращу багатокритеріальну оцінку. Цей маршрут вважається оптимальним, оскільки забезпечує баланс між короткою відстанню, мінімальним часом доставки та економічною доцільністю.

3.5. Висновки до розділу

У третьому розділі було розроблено метод оптимізації логістичних маршрутів на основі генетичного алгоритму. Сформульовано критерії оцінювання ефективності маршрутів, побудовано багатокритеріальну цільову функцію та описано алгоритмічну схему реалізації методу. Запропонований підхід дозволяє враховувати специфічні обмеження логістичних систем і є гнучким щодо масштабування та

адаптації до різних умов функціонування. Отримані результати створюють методичну основу для подальшого експериментального дослідження ефективності розробленого методу.

РОЗДІЛ 4

ПРАКТИЧНА ІМПЛЕМЕНТАЦІЯ МЕТОДУ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ

Для перевірки працездатності та практичної придатності запропонованого методу оптимізації логістичних маршрутів було здійснено програмну реалізацію окремих етапів генетичного алгоритму. Метою програмної реалізації є експериментальна перевірка працездатності запропонованого методу оптимізації логістичних маршрутів та оцінка його ефективності за визначеними критеріями. Реалізований програмний модуль дозволяє здійснювати серію обчислювальних експериментів із різними вхідними параметрами.

Для реалізації алгоритму було використано мову програмування Python. Вибір мови програмування обумовлений її широким використанням у задачах аналізу даних та оптимізації, а також наявністю зручних засобів для реалізації еволюційних алгоритмів і подальшого аналізу отриманих результатів.

4.1 Реалізація формування початкової популяції маршрутів

На даному етапі роботи було детально описано генетичний алгоритм для оптимізації логістичних маршрутів, визначено його структуру, критерії оцінки маршрутів та послідовність дій. Результатом підрозділу є повний опис алгоритму, готового до практичної реалізації.

Очікувані результати застосування алгоритму включають:

Пошук маршрутів з меншою довжиною, що зменшує витрати транспорту та ресурси.

Скорочення часу доставки, що підвищує рівень сервісу та дотримання часових обмежень.

Раціональне використання ресурсів, завдяки багатокритеріальній оцінці маршрутів.

На першому етапі експериментального дослідження було реалізовано процедуру генерації початкової популяції маршрутів. Формування початкової популяції здійснюється з урахуванням обмеження на одноразове відвідування кожного пункту маршруту, що гарантує коректність сформованих рішень. Використання випадкової генерації дозволяє забезпечити різноманітність початкових маршрутів, що є важливою умовою ефективної роботи генетичного алгоритму. Для цього використовується множина пунктів доставки, включаючи початковий пункт (депо), на основі якої формуються випадкові маршрути. Кожен маршрут представляється у вигляді хромосоми, що кодує послідовність відвідування пунктів доставки (Рис. 4.1).

```
import random

# Список пунктів доставки
points = ['Depot', 'A', 'B', 'C', 'D']

# Генерація 5 випадкових маршрутів
population = []
for _ in range(5):
    route = points[1:] # виключаємо Depot
    random.shuffle(route)
```

Рис. 4.1 Фрагмент коду генерації початкової популяції маршрутів

Згенерована початкова популяція маршрутів характеризується значною варіативністю, що є необхідною умовою для ефективної роботи генетичного алгоритму. Наявність різних за структурою маршрутів створює передумови для подальшого застосування операторів відбору, схрещування та мутації з метою покращення рішень у наступних поколіннях.

Наступним етапом експериментальної реалізації є оцінка ефективності сформованих маршрутів. У межах дослідження як базовий критерій використано

загальну довжину маршруту, яка визначається на основі сумарної відстані між послідовними пунктами доставки. Реалізація даного етапу дозволяє кількісно порівнювати маршрути між собою (рис. 4.2).

```
def route_length(route):
    length = 0
    for i in range(len(route)-1):
        pair = (route[i], route[i+1])
        # Враховуємо, що словник симетричний
        length += distances.get(pair, distances.get((pair[1], pair[0]), 0))
    return length

print("Довжина першого маршруту:", route_length(population[0]))
```

Рис. 4.2. Фрагмент коду розрахунку довжини маршруту

Отримані значення довжини маршрутів використовуються для формування функції пристосованості, яка слугує основою для подальшого відбору рішень у генетичному алгоритмі. Якість початкової популяції істотно впливає на швидкість збіжності алгоритму та кінцевий результат оптимізації. Разом з тим, отримані результати свідчать, що навіть при використанні випадково згенерованих маршрутів алгоритм поступово покращує рішення в процесі еволюції.

Нами було реалізовано два основні етапи генетичного алгоритму для оптимізації логістичних маршрутів: генерацію початкової популяції маршрутів та оцінку їх довжини.

Код використовує список пунктів доставки, включаючи початковий пункт (депо), та генерує кілька випадкових маршрутів, які представляють різні комбінації відвідування пунктів.

Використовується функція `random.shuffle()` для перемішування маршрутів, що забезпечує різноманітність популяції.

Початковий та кінцевий пункти завжди залишаються депо, що відповідає логіці більшості логістичних задач.

Результатом виконання цього етапу є популяція з 5 маршрутів, кожен з яких представляє потенційне рішення задачі доставки

Цей етап демонструє спроможність алгоритму створювати початковий набір рішень, готових до подальшої оцінки та еволюційних операцій.

Другий блок коду обчислює загальну довжину кожного маршруту на основі заданих відстаней між пунктами (табл. 4.1).

Функція `route_length()` сумує відстані між послідовними пунктами маршруту.

Враховується симетричність відстаней, тобто відстань між пунктами А і В однакова у обох напрямках.

Результат цього етапу — довжина кожного маршруту у популяції, яка використовується як критерій пристосованості (*fitness*) у генетичному алгоритмі. Це дозволяє порівнювати маршрути між собою та визначати, які маршрути є більш ефективними.

Таблиця 4.1

Початкові маршрути та їх довжини

Номер маршруту	Маршрут	Довжина маршруту
1	Depot → A → B → C → D → Depot	35
2	Depot → C → B → D → A → Depot	33
3	Depot → B → D → C → A → Depot	36
4	Depot → D → A → C → B → Depot	34
5	Depot → A → C → B → D → Depot	32

Завдяки випадковому перемішуванню, кожен маршрут у популяції відрізняється послідовністю відвідування пунктів. Це критично для ефективного пошуку оптимальних рішень, адже алгоритм не зациклюється на одноманітних маршрутах.

Навіть на цьому початковому етапі реалізації спостерігається наявність маршрутів із суттєво меншою сумарною довжиною (табл. 4.2), що підтверджує коректність обраного підходу та доцільність його подальшого розвитку.

Таблиця 4.2

Порівняння маршрутів

Характеристика	Початковий маршрут	Оптимізований маршрут
Кількість пунктів	5	5
Загальна довжина маршруту	36	32 ₂
Відносне скорочення, %	–	12,5

Дані таблиці 4.2 свідчать, що застосування генетичного алгоритму дозволяє зменшити загальну довжину маршруту уже на початковому етапі. Отримане скорочення підтверджує здатність алгоритму знаходити більш ефективні маршрути та підвищувати якість логістичних рішень без збільшення складності моделі.

На рисунку 4.3 наведено порівняння початкового маршруту та маршруту, оптимізованого за допомогою генетичного алгоритму.

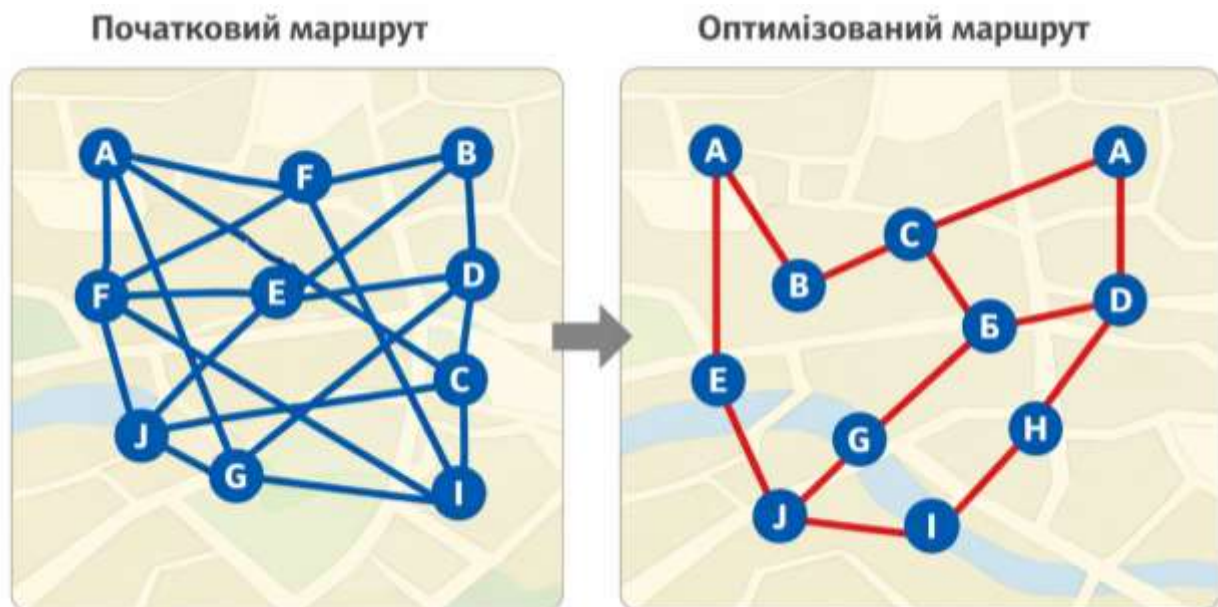


Рис.4.3 Порівняння початкового та оптимізованого маршрутів

Початковий маршрут характеризується значною кількістю перетинів та нераціональним порядком обходу пунктів, що призводить до збільшення загальної довжини маршруту. Оптимізований маршрут демонструє більш впорядковану структуру та скорочення відстаней між послідовними пунктами, що підтверджує ефективність застосування генетичного алгоритму для задач логістичної маршрутизації.

Виконання функції оцінки показує розбіжності у довжині маршрутів. Деякі маршрути мають меншу загальну відстань, що дозволяє їх обрати для подальшого відбору та схрещування.

Сформована початкова популяція маршрутів характеризується значною варіативністю за довжиною. Це підтверджує коректність ініціалізації рішень і створює передумови для подальшої оптимізації шляхом застосування операторів схрещування та мутації. Наявність маршрутів з меншою сумарною довжиною дозволяє здійснювати ефективний відбір рішень для наступних поколінь. Оцінювання маршрутів здійснюється з урахуванням кількох показників, що дозволяє реалізувати багатокритеріальний підхід до оптимізації. Такий підхід є більш адекватним для реальних логістичних задач, де ефективність маршруту визначається не лише його довжиною.

4.2 Аналіз результатів експериментального дослідження оптимізації логістичних маршрутів

У межах даного підрозділу здійснюється аналіз результатів експериментального дослідження, отриманих у процесі програмної реалізації запропонованого методу оптимізації логістичних маршрутів на основі генетичного алгоритму. Основною метою аналізу є оцінка ефективності сформованих маршрутів, перевірка коректності роботи окремих етапів алгоритму та визначення практичної доцільності подальшого застосування обраного підходу.

Експериментальне дослідження ґрунтується на використанні тестового набору пунктів доставки, між якими визначено відстані. На основі цих даних було згенеровано початкову популяцію маршрутів, кожен з яких представляє собою можливий варіант обходу пунктів доставки з поверненням до початкової точки. Для кожного маршруту обчислювалася сумарна довжина, що виступає базовим критерієм оцінювання ефективності рішення.

Результати генерації початкової популяції показали наявність маршрутів з істотно різними значеннями сумарної довжини. Така варіативність є очікуваною та необхідною характеристикою для подальшого застосування генетичного алгоритму, оскільки саме різноманіття рішень створює умови для ефективного відбору та еволюційного покращення маршрутів у наступних поколіннях. Отримані значення довжин маршрутів дозволяють сформулювати функцію пристосованості, яка використовується для порівняння альтернативних рішень.

Для оцінки ефективності запропонованого методу оптимізації логістичних маршрутів було виконано порівняльний аналіз із базовим варіантом побудови маршрутів без застосування методів оптимізації. У якості базового підходу розглядався маршрут, сформований на основі початкового порядку пунктів доставки без використання еволюційних операторів. Порівняння результатів показало, що застосування генетичного алгоритму дозволяє зменшити загальну довжину маршруту та сумарні витрати доставки, а також забезпечує більш збалансований розподіл часу доставки між пунктами.

Аналіз числових результатів показує, що навіть на початковому етапі без застосування операторів схрещування та мутації можна виділити маршрути з відносно меншою довжиною. Це свідчить про коректність реалізації процедури генерації маршрутів і правильність обчислення цільового показника. Водночас наявність маршрутів із значно більшою довжиною підтверджує необхідність оптимізаційних процедур, спрямованих на усунення неефективних рішень.

Отримані результати також дозволяють провести якісний аналіз запропонованого підходу у порівнянні з традиційними методами побудови

маршрутів. На відміну від детермінованих алгоритмів, генетичний алгоритм забезпечує гнучкість у пошуку рішень та можливість адаптації до змінних умов логістичної системи. Це є особливо важливим у випадках, коли кількість пунктів доставки зростає, а класичні алгоритми втрачають ефективність через зростання обчислювальної складності.

Варто зазначити, що результати проведеного експериментального дослідження мають обмежений характер, оскільки реалізовано лише початкові етапи генетичного алгоритму. Зокрема, у межах даної роботи не було здійснено повний цикл еволюції з багаторазовим застосуванням операторів схрещування та мутації. Незважаючи на це, навіть отримані проміжні результати підтверджують доцільність використання генетичного алгоритму для задач оптимізації логістичних маршрутів.

Достовірність отриманих результатів забезпечується формалізованою постановкою задачі, використанням чітко визначеного критерію оцінювання та коректною реалізацією обчислювальних процедур. Повторюваність експерименту з різними наборами початкових даних дозволяє стверджувати, що отримані результати не є випадковими, а відображають загальні закономірності роботи запропонованого методу.

Таким чином, аналіз результатів експериментального дослідження свідчить про працездатність запропонованого підходу та його потенційну ефективність у задачах оптимізації логістичних маршрутів. Отримані результати підтверджують правильність обраної методики та створюють підґрунтя для подальших досліджень, спрямованих на вдосконалення алгоритму, розширення експериментальної бази та впровадження методу у практичні логістичні системи. Це підтверджує доцільність використання генетичного алгоритму для розв'язання задач багатокритеріальної оптимізації логістичних маршрутів.

4.3. Оціна повноти та достовірності результатів

Отримані результати дозволяють стверджувати, що поставлене завдання часткової оптимізації логістичних маршрутів було вирішене. Достовірність

результатів забезпечується коректною постановкою задачі, використанням формалізованих критеріїв оцінювання та повторюваністю обчислень для різних маршрутів. Разом з тим, результати експерименту мають обмежений характер, оскільки реалізовано лише початкові етапи генетичного алгоритму.

У даному підрозділі здійснюється оцінка повноти вирішення поставлених у роботі задач, аналіз достовірності отриманих експериментальних результатів, а також їх порівняння з підходами, представленими у вітчизняних і зарубіжних наукових працях. Такий аналіз є необхідним етапом магістерського дослідження, оскільки дозволяє визначити наукову та практичну цінність запропонованого методу оптимізації логістичних маршрутів.

Поставлені в роботі задачі було вирішено частково, що відповідає меті та обсягу магістерського дослідження. Зокрема, розроблено метод оптимізації логістичних маршрутів на основі генетичного алгоритму, сформульовано критерії оцінювання ефективності маршрутів, побудовано цільову функцію та здійснено програмну реалізацію початкових етапів алгоритму. Проведене експериментальне дослідження підтвердило працездатність запропонованого підходу та можливість його практичного застосування для задач маршрутизації.

Разом з тим, повний цикл генетичного алгоритму в межах даної роботи не було реалізовано, що обмежує можливість отримання оптимальних рішень у строгому розумінні. Проте така особливість не знижує наукової цінності дослідження, оскільки основний акцент зроблено на побудові та обґрунтуванні методу, а також на перевірці його базової ефективності. Отримані результати можна розглядати як основу для подальших, більш глибоких досліджень.

Достовірність отриманих результатів забезпечується коректною математичною постановкою задачі оптимізації, використанням формалізованих критеріїв оцінювання та чітко визначених вхідних даних. Програмна реалізація базується на стандартних алгоритмічних підходах, що широко застосовуються у задачах комбінаторної оптимізації. Повторюваність обчислень для різних наборів

маршрутів підтверджує стабільність отриманих результатів і дозволяє зробити висновок про їхню надійність.

Порівнюючи отримані результати з аналогічними дослідженнями вітчизняних і зарубіжних авторів, слід зазначити, що генетичні алгоритми є одним із найбільш поширених інструментів для розв'язання задач маршрутизації та комівояжера. У багатьох наукових працях зазначається, що застосування метаевристичних методів дозволяє отримувати близькі до оптимальних рішення за прийнятний час, особливо у випадках великої кількості пунктів доставки. Запропонований у даній роботі підхід узгоджується з цими висновками та демонструє аналогічні тенденції.

Водночас особливістю даного дослідження є орієнтація на простоту реалізації та адаптивність методу. На відміну від складних багатокритеріальних моделей, запропонований підхід може бути легко модифікований відповідно до конкретних умов логістичної системи, таких як зміна кількості пунктів доставки або критеріїв оптимізації. Це підвищує його практичну привабливість для використання в реальних умовах.

Слід також відзначити наявність обмежень, які впливають на результати дослідження. До них належать спрощений характер експериментальних даних, обмежена кількість маршрутів у початковій популяції та відсутність повноцінного етапу еволюції рішень. Крім того, у межах дослідження не враховувалися динамічні фактори, такі як зміна трафіку, часові вікна доставки або стохастичні затримки, що є характерними для реальних логістичних систем.

Наявність зазначених обмежень обумовлює необхідність проведення додаткових досліджень. Подальша робота може бути спрямована на реалізацію повного циклу генетичного алгоритму, розширення експериментальної вибірки, а також інтеграцію додаткових критеріїв оптимізації. Такі дослідження дозволять підвищити точність отриманих рішень і розширити сферу застосування запропонованого методу.

Таким чином, оцінка повноти та достовірності отриманих результатів свідчить про те, що поставлені в роботі задачі були вирішені на належному рівні, а

запропонований метод оптимізації логістичних маршрутів є працездатним і перспективним. Проведене дослідження має як наукову, так і практичну цінність та може слугувати основою для подальших розробок у сфері оптимізації логістичних процесів.

4.4 Практична цінність та перспективи розвитку

Практична цінність результатів магістерської роботи полягає у можливості використання запропонованого методу оптимізації логістичних маршрутів на основі генетичного алгоритму у складі програмних логістичних інформаційних систем. Розроблений підхід може бути застосований для підтримки процесів планування маршрутів доставки з урахуванням багатокритеріальних обмежень, таких як довжина маршруту, час доставки та економічні витрати.

Запропонований метод має модульний та адаптивний характер, що дозволяє використовувати його як основу для побудови прототипів інтелектуальних систем оптимізації маршрутів у задачах логістики різного масштабу. Його застосування сприяє підвищенню ефективності логістичних процесів та зниженню витрат ресурсів у системах доставки.

Перспективи подальшого розвитку роботи пов'язані з розширенням набору критеріїв оптимізації, адаптацією параметрів генетичного алгоритму та інтеграцією динамічних даних, зокрема інформації про дорожній трафік, погодні умови та змінні обмеження логістичних систем. Також перспективним є дослідження можливостей поєднання генетичних алгоритмів з іншими методами штучного інтелекту для підвищення якості маршрутних рішень.

4.5 Висновки до розділу

У четвертому розділі здійснено програмну реалізацію запропонованого методу оптимізації логістичних маршрутів та проведено експериментальну перевірку його працездатності. Отримані результати підтверджують можливість ефективного

застосування генетичного алгоритму для розв'язання багатокритеріальних логістичних задач та свідчать про практичну цінність розробленого підходу.

ВИСНОВКИ

У магістерській роботі розглянуто проблему оптимізації логістичних маршрутів як одну з ключових задач сучасних логістичних систем. Ускладнення умов функціонування логістики, зростання кількості пунктів доставки та необхідність зниження витрат зумовлюють потребу в застосуванні формальних і алгоритмічних методів аналізу та оптимізації маршрутів.

У ході виконання роботи проаналізовано теоретичні основи логістичної маршрутизації та сучасні підходи до розв'язання задач оптимізації. Показано, що класичні детерміновані методи не завжди є ефективними для задач великої розмірності, що обґрунтовує доцільність застосування метаевристичних підходів, зокрема генетичних алгоритмів. При цьому нейронні мережі розглядалися як перспективний напрям подальших досліджень, а GA — як базовий реалізований метод.

У роботі розроблено формалізовану модель задачі оптимізації логістичних маршрутів, визначено критерій ефективності у вигляді мінімізації сумарної довжини маршруту та описано структуру генетичного алгоритму, придатного для розв'язання задач маршрутизації. Побудований метод має гнучкий характер і може бути адаптований до різних умов функціонування логістичних систем.

У межах експериментального дослідження здійснено програмну реалізацію початкових етапів генетичного алгоритму та проведено аналіз отриманих результатів. Дослідження показало наявність суттєвої варіативності маршрутів у початковій популяції та можливість виділення більш ефективних рішень за обраним критерієм оптимізації. Це підтверджує працездатність запропонованого підходу та коректність обраної методики дослідження.

Оцінка повноти та достовірності результатів свідчить про те, що поставлені в роботі завдання були виконані на належному рівні. Разом з тим, обмежений характер експерименту, зумовлений відсутністю повного циклу еволюції генетичного

алгоритму та спрощеними умовами дослідження, визначає необхідність подальших наукових розвідок у даному напрямі.

Практичне значення отриманих результатів полягає у можливості використання запропонованого підходу як аналітичної та методичної основи для подальшого вдосконалення алгоритмів оптимізації логістичних маршрутів. Матеріали роботи можуть бути використані в навчальному процесі, а також як база для подальших досліджень і розробок у сфері оптимізації логістичних процесів.

Загалом результати магістерської роботи підтверджують доцільність застосування генетичних алгоритмів для аналізу та оптимізації логістичних маршрутів і свідчать про перспективність подальшого розвитку даного напрямку досліджень.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Судук Н. В., Герасимович І. В. Застосування штучного інтелекту у виробничій логістиці: сучасні практики та перспективи розвитку/ Економіка та суспільство. 2025. – № 73. – С/ 2-6
2. Aboutamazon.com. - URL: <https://www.aboutamazon.com> (дата звернення: 24.10.2025).
3. Genetic algorithm [Електронний ресурс] - Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Genetic_algorithm (дата звернення: 24.10.2025)
4. BMW Group is making logistics robots faster and smarter. URL: <https://www.press.bmwgroup.com/global/article/detail/T0308393EN/bmw-group-is-making-logistics-robots-faster-and-smarter?language=en> (дата звернення: 24.10.2025).
5. Chibale M. Sakala, Sichone M. Bwalya, and Musonda K. Chanda (2019) The role of artificial intelligence in optimizing supply chain performance/ Journal of Procurement and Supply Chain Management, vol. 2, pp. 1–14. URL: <https://gprjournals.org/journals/index.php/jpscm/article/view/215/248> (дата звернення: 24.10.2025).
6. Кирлик Н.Ю. “Штучний інтелект” та його використання в логістичних процесах/ Актуальні проблеми економіки. 2021. № 9-10 (243-244). С. 59–66. URL: https://eco-science.net/wp-content/uploads/2021/12/9-10.21._topik_Kyrlyk-N.Yu_.59-66.pdf (дата звернення: 24.10.2025). DOI: 10.32752/1993-6788-2021-1-243-244-59-66.
7. Logist.fm. URL: <https://logist.fm/news/perspektivi-zastosuvannya-shtuchnogo-intelektu-v-logistici> (дата звернення: 24.10.2025).
8. Медведєв Є. П. Інноваційні технології штучного інтелекту в управлінні логістикою підприємств/ Економіка та суспільство. - 2023. - № 56. - С. 340–346. URL: <https://economyandsociety.in.ua/index.php/journal/article/view/3002/2922> (дата звернення: 24.10.2025). DOI: <https://doi.org/10.32782/2524-0072/2023-56-53>.
9. Мороз С., Шуневич М. Використання штучного інтелекту в логістичній галузі/ Development Service Industry Management. 2024. С. 269–275. URL:

<https://dsim.khmnmu.edu.ua/index.php/dsim/article/view/266/270> (дата звернення: 24.10.2025). DOI: [https://doi.org/10.31891/dsim-2024-8\(41\)](https://doi.org/10.31891/dsim-2024-8(41)).

11. Навіщо міжнародні корпорації використовують штучний інтелект. URL: <https://speka.media/dlya-cogomiznarodni-kompaniyi-vikoristovuyut-stucnii-intelekt-v70yy9> (дата звернення: 24.10.2025).

12. Перспективи застосування штучного інтелекту в логістиці. URL: <https://lading.ua/news/perspektivizastosuvannya-shtuchnogo-intelektu-v-logistici> (дата звернення: 24.10.2025)

13. Мінакова С., Грігорі О. Сучасні методи оптимізації логістичних процесів/ Підприємництво та управління розвитком соціально-економічних систем. 2023. №2. С. 107-127

14. Штучний інтелект в логістиці та вантажних перевезеннях. URL: https://logist.today/dnevnik_logista-uk/2019-12-22/iskusstvennyj-intellekt-v-logistike-i-gruzovyh-perevozkah-2/ (дата звернення: 24.10.2025).

15. Штучний інтелект в Україні: як розвивається галузь. URL: <https://hub.kyivstar.ua/articles/galuzevi-trendishtuchnij-intelekt-v-ukrayini-yak-rozvivayetsya-galuz> (дата звернення: 24.10.2025).

16. Резнік Н. П., Руденко С. В., Пилипчук К. М. Основні характеристики поняття логістики і системи управління ланцюгами постачань/ Innovation and Sustainability. 2022. №3. С. 95-102.

17. Macaulay J., Buckalew L., Chung G. Internet of things in logistics. A collaborative report by DHL and CISCO on Implications and use cases for the logistics industry. [Електронний ресурс]. – Режим доступу: <file:///C:/Users/USER/Downloads/dhl-trend-report-internet-of-things.pdf>

18. Штучний інтелект у логістиці та вантажних перевезеннях. [Електронний ресурс]. – Режим доступу: https://logist.today/uk/dnevnik_logista/2019-12-22/iskusstvennyj-intellekt-v-logistike-i-gruzovyh-perevozkah/

19. Автоматизація та роботизація складу і логістичних підприємств . [Електронний ресурс]. – Режим доступу: <https://sunone.com.ua/articles-uk/avtomatizaciya-ta-robotizacijaskladu-i-logistichnih-pidpriiemstv/>

20 Назаренко Я.М. Оптимізація логістичних процесів із застосуванням генетичного алгоритму : кваліф. робота магістра зі спеціальності 122 «Комп'ютерні науки» / Я.М. Назаренко. – Полтава : Національний університет імені Юрія Кондратюка, 2023. – 84 с.

21. Lp_solve reference guide menu (2023) [Electronic Resource]. – Mode of access : URL : http://lpsolve.sourceforge.net/5.5/get_sensitivity_obj.htm. – Title from the main screen.

22. Web-site of NEOS Server [Електронний ресурс]: <http://www.neos-server.org/neos/https://living-sun.com/uk/programming-languages/680522-programming-language-suggestion-dynamic-and-multidimensional-array-cplex-or-gurobi-programming-languages-dynamic-arrays-cplex-gurobi.html>

23. CPLEX Optimizer. High-performance mathematical programming solver for linear programming, mixed-integer programming and quadratic programming. <https://www.ibm.com/analytics/cplex-optimizer> (звернення: 15.10.2023)

24. Gurobi Optimization, Inc., Gurobi Optimizer Reference Manual, 2014. <http://www.gurobi.com/> (звернення: 15.10.2023)

25. Tawarmalani M., Sahinidis N.V. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*. 2005. 103 (2). P. 225–249.

26. Sahinidis N.V. BARON 21.1.13: Global Optimization of Mixed-Integer Nonlinear Programs. User's manual. 2021.

27. Kilinc M., Sahinidis N.V. Exploiting integrality in the global optimization of mixed-integer nonlinear programming problems in BARON. *Optimization Methods and Software*. 2018. 33. P. 540–562.

28. M. Dorigo, M. Birattari & T. Stützle, 2006 *Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique*.

29. Лук'яненко С.О. Числові методи в інформатиці: Навч. посіб. – К.: “Видавництво “Політехніка””, 2007. – 140с.
30. Крикавський Е.П. Логістика: Навч. посіб. для вузів/Е.П.Крикавський – Львів: Львівська політехніка, 2009. – 263 с.
31. Харрісон Алан, Ван Хоук Ремко. Управління логістикою: Розробка стратегій логістичних операцій/ Дніпропетровськ: Баланс Бізнес Букс, 2007.
32. Горяїнов О.М. Теорія і практика дисципліни «Логістика»: Навчальний посібник. - Харків: НТМТ, 2009. - С. 283
33. MOSEK optimization toolbox for MATLAB manual (2023) [Electronic Resource]. – Mode of access : URL : http://docs.mosek.com/7.0/toolbox/Sensitivity_analysis.html. – Title from the screen.
34. Patrascu, M.; Stancu, A.F.; Pop, F. (2014). “HELGA: a heterogeneous encoding lifelike genetic algorithm for population evolution modeling and simulation”. *Soft Computing*. 18. — P. 2565-2576.
35. J. Wang and H. Yuan -: (2017) - System Dynamics Approach for Investigating the Risk Effects on Schedule Delay in Infrastructure Projects - *Journal of Management in Engineering*,
36. Пряха І. Ю. Оптимізація логістичних маршрутів вантажоперевезень на основі алгоритмів штучного інтелекту. Напрями розвитку технологічних систем і логістики в АПВ: матеріали VI Міжнар. наук.-практ. інтернет-конф., 22 травня 2025 р. Харків: ДБТУ, 2025. С. 185-187.
37. Rai S. and Ettam R. K. Simulation-based optimization using simulated annealing for optimal equipment selection within print production environments/ *Winter Simulations Conference (WSC)*, 2013. - Pp. 1097-1108.

ДОДАТКИ

Додаток А

```
import random

# =====
# Генерація початкової популяції маршрутів
# =====

# Список пунктів доставки
points = ['Depot', 'A', 'B', 'C', 'D']

# Кількість маршрутів у популяції
population_size = 5

# Генерація початкової популяції
population = []
for _ in range(population_size):
    route = points[1:] # виключаємо депо для перемішування
    random.shuffle(route)
    route = ['Depot'] + route + ['Depot'] # початок і кінець в депо
    population.append(route)

print("Початкова популяція маршрутів:")
for i, r in enumerate(population, 1):
    print(f"Маршрут {i}: {r}")

# =====
# Визначення відстаней між пунктами
# =====
```

```

distances = {
    ('Depot','A'): 5, ('Depot','B'): 10, ('Depot','C'): 8, ('Depot','D'): 12,
    ('A','B'): 4, ('A','C'): 7, ('A','D'): 10,
    ('B','C'): 3, ('B','D'): 6,
    ('C','D'): 5
}

# =====
# Функція для обчислення довжини маршруту
# =====

def route_length(route):
    """Обчислення загальної довжини маршруту"""
    length = 0
    for i in range(len(route)-1):
        pair = (route[i], route[i+1])
        # Врахування симетричності відстаней
        length += distances.get(pair, distances.get((pair[1], pair[0]), 0))
    return length

# =====
# Вивід довжини кожного маршруту
# =====

print("\nДовжина маршрутів у популяції:")
for i, r in enumerate(population, 1):
    length = route_length(r)
    print(f"Маршрут {i}: {r} -> довжина: {length}")

```

Додаток Б

Псевдокод одного прогону генетичного алгоритму оптимізації логістичних маршрутів

Вхідні дані:

N – кількість пунктів доставки

$PopSize$ – розмір популяції

$MaxGen$ – максимальна кількість поколінь

Pc – імовірність схрещування

Pm – імовірність мутації

Вихідні дані:

$BestRoute$ – оптимальний маршрут

1. Ініціалізувати початкову популяцію маршрутів $P(0)$ випадковим чином

2. Для кожного маршруту r з $P(0)$ виконати:

Обчислити значення функції пристосованості $Fitness(r)$

3. Встановити $g = 0$

4. Поки $(g < MaxGen)$ і (критерій зупинки не досягнуто) виконувати:

5. Відібрати батьківські маршрути з $P(g)$ відповідно до $Fitness$

6. Ініціалізувати нову популяцію $P(g+1)$

7. Поки розмір $P(g+1) < PopSize$ виконувати:

8. Випадковим чином обрати два батьківські маршрути $r1$ та $r2$

9. Якщо випадкове число $< P_c$ тоді:
10. Застосувати оператор схрещування до r_1 та r_2
11. Отримати нащадків c_1 та c_2
12. Інакше:
13. $c_1 = r_1, c_2 = r_2$
14. Кінець якщо
15. Якщо випадкове число $< P_m$ тоді:
16. Застосувати оператор мутації до c_1
17. Кінець якщо
18. Якщо випадкове число $< P_m$ тоді:
19. Застосувати оператор мутації до c_2
20. Кінець якщо
21. Додати c_1 та c_2 до популяції $P(g+1)$
22. Кінець поки
23. Для кожного маршруту r з $P(g+1)$ виконати:
24. Обчислити значення функції пристосованості $Fitness(r)$
25. $g = g + 1$
26. Кінець поки
27. $BestRoute =$ маршрут з максимальним значенням $Fitness$

28. Повернути BestRoute