

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій  
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)  
Розроблення логістичної системи для побудови оптимального маршруту  
переміщення вантажів на підприємстві  
(тема)

Виконав:

здобувач II-го року навчання,  
групи КІТПВм-23-1

Дідик П.Ю.

(прізвище, ініціали)

Спеціальність 174 Автоматизація та  
комп'ютерно-інтегровані технології  
(код і повна назва спеціальності)

Тип програми освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерно-інтегровані  
технологічні процеси і виробництва  
(повна назва освітньої програми)

Керівник доц. Максимова С.С.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

(підпис)

Невлюдов І.Ш.

(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ АКТ \_\_\_\_\_  
Кафедра \_\_\_\_\_ КІТАР \_\_\_\_\_  
Рівень вищої освіти \_\_\_\_\_ Другий (магістерський) \_\_\_\_\_  
Спеціальність \_\_\_\_\_ 174 Автоматизація та комп'ютерно-інтегровані технології \_\_\_\_\_  
(код і повна назва)  
Тип програми \_\_\_\_\_ Освітньо-професійна \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)  
Освітня програма \_\_\_\_\_ Комп'ютерно-інтегровані технологічні процеси і виробництва \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р.

**ЗАВДАННЯ**  
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві \_\_\_\_\_ Дідику Павлу Юрійовичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

- Тема роботи \_\_\_\_\_ Розроблення логістичної системи для побудови оптимального маршруту переміщення вантажів на підприємстві \_\_\_\_\_  
затверджена наказом університету від \_\_\_\_\_ 22 листопада 2024 р. № \_\_\_\_\_ 1231Ст
- Термін подання здобувачем роботи до екзаменаційної комісії \_\_\_\_\_ 31 січня \_\_\_\_\_ 2025 р.
- Вихідні дані до роботи \_\_\_\_\_ Розробити логістичну систему для побудови оптимального маршруту переміщення вантажів на підприємстві. Використовувати Windows ОС, середовище розробки PyCharm, БД MySQL мову програмування Python \_\_\_\_\_  
4.1 Вступ 4.2 Аналіз теоретичних основ та сучасного стану проблеми 4.3 Проектування логістичної системи на виробництві 4.4 Розроблення програми 4.5 Експериментальне дослідження 4.6 Охорона праці 4.7 Висновки 4.8 Перелік джерел посилання.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій Демонстраційний матеріал представлений у форматі презентації PowerPoint (\*.ppt) – 10 с. формату А4

6. Консультанти розділів роботи

Найменування розділу	Керівник (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів	Примітка
Пор.	Назва етапів кваліфікаційної роботи	Термін виконання етапів	Примітка
1.	<i>Отримання завдання кваліфікаційної роботи</i>	25.11.2024	Виконано
2.	<i>Аналіз завдання та пошук інформації</i>	02.12.2024	Виконано
3.	<i>Вибір методів та інструментів до розробки алгоритму</i>	07.12.2024	Виконано
4.	<i>Розробка алгоритму програми</i>	10.12.2024	Виконано
5.	<i>Вибір мови програмування та середовища розробки</i>	12.12.2024	Виконано
6.	<i>Розробка програми</i>	25.12.2024	Виконано
7.	<i>Тестування програми</i>	27.12.2024	Виконано
8.	<i>Аналіз отриманих результатів та оцінка</i>	30.12.2024	Виконано
9.	<i>Розробка «Посібника користувача»</i>	02.01.2025	Виконано
10.	<i>Оформлення пояснювальної записки та програмної</i>	10.01.2025	Виконано
11.	<i>Редагування графічного вмісту презентаційних матеріалів</i>	12.01.2025	Виконано
12.	<i>Представлення на рецензування</i>	15.01.2025	Виконано
13.	<i>Представлення кваліфікаційної роботи</i>	20.01.2025	Виконано

Дата видачі завдання 25 листопада 2024 р.

Здобувач \_\_\_\_\_ Дідик П.Ю.  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Максимова С.С.  
(підпис) (посада, прізвище, ініціали)

*Я як студент ХНУРЕ розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.*

*20 січня 2025 р.*



*Дідик П.Ю*

## РЕФЕРАТ

Пояснювальна записка: 70 с., 9 рис., 1 формула., 4 дод., 30 джерел.

### ВАНТАЖ, ЛОГІСТИЧНА СИСТЕМА, КЛАСТЕРИЗАЦІЯ, МАРШРУТ, ПЕРЕВЕЗЕННЯ ВАНТАЖІВ, АВТОМАТИЗАЦІЯ

Об'єкт дослідження – процес переміщення вантажів.

Предмет дослідження – логістична система для побудови оптимального маршруту.

Мета дослідження – підвищення ефективності переміщення вантажів на підприємстві за рахунок розробки логістичної системи.

Кваліфікаційна робота присвячена розробці логістичної системи, яка дозволяє оптимізувати процес перевезення вантажів. У дослідженні проаналізовано сучасні методи кластеризації, що забезпечують ефективне групування географічних точок на початкових етапах організації транспортної логістики. Використання кластеризації дозволяє визначити ключові вузлові напрямки, які стають основою для побудови оптимальних маршрутів перевезення.

У роботі впроваджено методи автоматизації для інтеграції кластеризаційних даних із системами побудови маршрутів, що мінімізує витрати, скорочує час доставки та підвищує загальну ефективність логістичних процесів. Виконано моделювання логістичних процесів у програмному середовищі, що дало змогу перевірити працездатність розробленої системи на реальних даних підприємства.

Результати дослідження включають алгоритм побудови оптимального маршруту з урахуванням географічних та економічних факторів. Запропонована система забезпечує можливість масштабування для використання на підприємствах із різними масштабами діяльності.

Робота має практичне значення для підприємств, які прагнуть зменшити логістичні витрати та підвищити точність і надійність доставки вантажів.

Рекомендації щодо застосування: результати дослідження можуть бути впроваджені на підприємствах, які займаються перевезенням вантажів, для автоматизації логістичних процесів та зменшення експлуатаційних витрат.

Результати кваліфікаційної роботи апробовані у всеукраїнській конференції та у збірнику наукових робіт категорії „Б” Переліку наукових фахових видань України.

Також, отримані результати роботи можна віднести до цілі сталого розвитку «Промисловість, інновації та інфраструктура». Завдання 9.4. Сприяти прискореному розвитку високо- та середньовисокотехнологічних секторів переробної промисловості, які формуються на основі використання ланцюгів «освіта – наука – виробництво» та кластерного підходу за напрямками: розвиток інноваційної екосистеми; розвиток інформаційно-телекомунікаційних технологій (ІКТ); застосування ІКТ в АПК, енергетиці, транспорті та промисловості; високотехнологічне машинобудування; створення нових матеріалів; розвиток фармацевтичної та біоінженерної галузей.

## **ABSTRACT**

Explanatory Note: 70 pages, 9 figures, 1 formula, 4 references, 30 sources.

**CARGO, LOGISTICS SYSTEM, CLUSTERING, ROUTE, CARGO TRANSPORTATION, AUTOMATION.**

Object of research – the process of cargo transportation.

Subject of research – a logistics system for building optimal routes.

Purpose of research – to improve the efficiency of cargo transportation within an enterprise by developing a logistics system.

This qualification work is dedicated to the development of a logistics system that optimizes the cargo transportation process. The study analyzes modern clustering methods, which enable efficient grouping of geographic points at the initial stages of organizing transport logistics. The use of clustering allows the identification of key nodal directions, forming the foundation for constructing optimal transportation routes.

The work incorporates automation methods to integrate clustering data with routing systems, minimizing costs, reducing delivery time, and improving the overall efficiency of logistics processes. Modeling of logistics processes was conducted in a software environment, enabling the verification of the developed system using real enterprise data.

The research results include an algorithm for constructing optimal routes, taking into account geographic and economic factors. The proposed system offers scalability for enterprises of various operational scales.

This work is of practical significance for enterprises aiming to reduce logistics costs and enhance the accuracy and reliability of cargo delivery.

Recommendations for application: The research results can be implemented in enterprises involved in cargo transportation to automate logistics processes and reduce operational costs.

The results of the qualification work were approved at the All-Ukrainian conference and in the collection of scientific works of category "B" of the List of Scientific Professional Publications of Ukraine.

The results of this work can also be attributed to the Sustainable Development Goal "Industry, Innovation, and Infrastructure." Task 9.4: Promote accelerated development of high- and medium-high-tech sectors of the manufacturing industry, based on the use of "education – science – production" chains and a cluster-based approach in areas such as the development of the innovation ecosystem, information and communication technologies (ICT), ICT application in agriculture, energy, transportation, and industry, high-tech mechanical engineering, creation of new materials, and development of the pharmaceutical and bioengineering industries.

## ЗМІСТ

Перелік умовних позначень та скорочень.....	11
Вступ .....	13
1 Аналіз теоретичних основ та сучасного стану проблеми.....	15
1.1 Основи транспортної логістики на виробництві .....	15
1.2 Сучасні підходи до кластеризації даних у логістиці.....	16
1.3 Технології автоматизації логістичних процесів .....	19
1.4 Аналіз існуючих методів побудови маршрутів та вибір кластеризаційних методів для організації логістичних процесів .....	20
1.5 Аналіз програмного забезпечення для моделювання логістичних систем.....	25
1.6 Висновки до першого розділу .....	29
2 Проектування логістичної системи на виробництві .....	30
2.1 Постановка задачі .....	30
2.2 Розробка алгоритму побудови оптимальних маршрутів на виробництві .....	31
2.3 Інтеграція кластеризаційних даних в логістичну систему .....	34
2.4 Використання методів машинного навчання для оптимізації логістичних процесів.....	38
2.5 Застосування теорії автоматичного управління .....	40
2.6 Висновки до другого розділу.....	41
3 Розроблення програми .....	43
3.1 Застосування методів кластеризації.....	43
3.2 Візуалізація отриманих даних .....	47
3.3 Засоби для побудови маршрутів на виробництві. Візуалізація маршрутів. ....	51
3.4 Висновки до третього розділу .....	55

4 Експериментальне дослідження.....	56
4.1 Моделювання логістичних процесів у програмному середовищі ..	56
4.2 Перевірка працездатності системи на реальних даних підприємства .....	58
4.3 Аналіз результатів експерименту .....	59
4.4 Рекомендації для застосування програми та можливі варіанти покращення.....	60
4.5 Висновки до четвертого розділу .....	61
5 Охорона праці .....	63
5.1 Аналіз умов праці на транспортних підприємствах .....	63
5.2 Рекомендації щодо поліпшення умов праці.....	64
Висновки.....	65
Перелік джерел посилання.....	66
Додаток А Посібник користувача .....	71
Додаток Б Апробація результатів кваліфікаційної роботи .....	82
Додаток В Висвітлення результатів кваліфікаційної роботи.....	92
Додаток Г Текст програми.....	108

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

AI (Artificial Intelligence) – штучний інтелект, який аналізує дані та оптимізує логістичні процеси;

AMR (Autonomous Mobile Robots) – автономні мобільні роботи, які виконують операції на складах;

API (Application Programming Interface) – інтерфейс для взаємодії між програмними компонентами;

BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) – ієрархічний алгоритм кластеризації, орієнтований на обробку великих наборів даних із мінімальними витратами пам'яті;

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) – алгоритм кластеризації на основі щільності, який дозволяє виділяти кластери будь-якої форми, ігноруючи шумові дані;

GUI (Graphical User Interface) – графічний інтерфейс користувача;

H3 (Hexagonal Hierarchical Geospatial Indexing System) – геопросторовий алгоритм кластеризації, який використовує шестикутні сітки для ієрархічної організації даних;

IDE (Integrated Development Environment) – інтегроване середовище розробки, що забезпечує зручний інтерфейс для написання та тестування програм

IoT (Internet of Things) – інтернет речей, що об'єднує пристрої для збору та аналізу даних;

k-means (k-means clustering algorithm) – алгоритм кластеризації, який розділяє дані на задану кількість кластерів шляхом мінімізації розсіювання всередині кожного кластеру;

NumPy (Numerical Python) – бібліотека Python для виконання наукових розрахунків і роботи з багатовимірними масивами;

RFID (Radio Frequency Identification) – технологія ідентифікації за допомогою радіочастот;

SciPy (Scientific Python) – бібліотека для виконання наукових і технічних розрахунків;

TMS (Transportation Management Systems) – системи управління транспортом, що дозволяють автоматизувати планування та моніторинг перевезень;

WMS (Warehouse Management Systems) – системи управління складом, які автоматизують операції з обробки та зберігання товарів.

## ВСТУП

У сучасних умовах розвитку промислових підприємств та глобальної торгівлі логістичні системи є ключовими елементами забезпечення ефективності економічних процесів. Оптимізація логістичних маршрутів сприяє зменшенню витрат, скороченню часу доставки, підвищенню точності та надійності транспортування вантажів. Зростання масштабів перевезень та підвищення вимог до їхньої організації створюють потребу у впровадженні інноваційних підходів у логістичну діяльність, що робить тему дослідження актуальною.

Об'єктом дослідження є процес транспортування вантажів. Предметом дослідження виступає логістична система для побудови оптимального маршруту перевезення. Метою роботи є підвищення ефективності транспортування вантажів на підприємствах шляхом розробки інтегрованої логістичної системи з використанням сучасних методів кластеризації та автоматизації.

У ході дослідження було проведено аналіз сучасних методів кластеризації, які дозволяють ефективно групувати географічні точки для подальшої організації маршрутів. На основі отриманих даних розробити алгоритм побудови оптимальних логістичних маршрутів із врахуванням географічних, економічних та екологічних факторів. Використання кластеризації на початкових етапах планування дозволяє визначити ключові вузли, що оптимізують логістичну мережу підприємства.

Наукова новизна роботи полягає у розробці інтегрованого підходу до оптимізації логістичних маршрутів, який базується на використанні кластеризації у поєднанні з алгоритмами машинного навчання. Запропонована система здатна адаптуватися до змін у вхідних даних, таких як обсяги перевезень чи географічні особливості. У роботі вперше реалізовано автоматизацію процесу інтеграції кластеризаційних даних із системами побудови маршрутів, що забезпечує підвищення ефективності транспортування вантажів.

Практична цінність роботи полягає у впровадженні розробленої системи в діяльність підприємств для скорочення експлуатаційних витрат, зменшення

негативного впливу на довкілля та підвищення конкурентоспроможності. Результати дослідження можуть бути масштабовані для застосування на підприємствах із різними рівнями логістичної активності.

Методи дослідження включають математичне моделювання, кластерний аналіз, автоматизацію процесів, аналіз даних та машинне навчання. Застосування методів машинного навчання дозволяє підвищити точність кластеризації географічних точок та оптимізації маршрутів шляхом навчання моделі на основі реальних даних підприємства. Це сприяє адаптивності системи до змін у транспортній мережі та вимогах підприємств.

Таким чином, розроблена логістична система є інноваційним підходом до організації перевезень, що забезпечує ефективність логістичних процесів та їхню інтеграцію в сучасні умови функціонування підприємств.

Пояснювальну записку виконати та оформити згідно [1] та [2].

# 1 АНАЛІЗ ТЕОРЕТИЧНИХ ОСНОВ ТА СУЧАСНОГО СТАНУ ПРОБЛЕМИ

## 1.1 Основи транспортної логістики на виробництві

Транспортна логістика – це ключовий елемент загальної логістичної системи, що забезпечує управління потоками вантажів, інформації та ресурсів у процесі їх переміщення. Основною метою транспортної логістики є забезпечення оптимальних умов доставки вантажів за критеріями часу, витрат, безпеки та якості.

Транспортна логістика є однією з найважливіших складових загальної логістичної системи підприємства. Вона включає планування, організацію, контроль і управління процесами переміщення матеріальних ресурсів від постачальника до споживача. Ця діяльність охоплює як внутрішнє транспортування в межах підприємства, так і зовнішню логістику, яка забезпечує міжрегіональне та міжнародне перевезення.

Згідно з дослідженнями в галузі логістики, інтеграція сучасних інформаційних систем дозволяє оптимізувати транспортні процеси та забезпечувати підвищення ефективності на 15–20% шляхом зменшення витрат на перевезення та покращення планування маршрутів [3]. Крім того, активне використання методів прогнозування дозволяє адаптуватися до змін попиту та мінімізувати ризики невиконання термінів доставки.

До основних принципів транспортної логістики належать:

- ефективність – мінімізація витрат і часу на транспортування при збереженні високої якості послуг;
- раціоналізація маршрутів – розробка оптимальних маршрутів перевезення для зменшення витрат і часу доставки;
- інтеграція – об'єднання транспортних потоків із іншими елементами логістики (складським, виробничим, інформаційним) для забезпечення безперервності процесу;

- гнучкість – можливість адаптації транспортних рішень до змін у зовнішньому середовищі, таких як попит, умови постачання чи транспортна інфраструктура;

- інноваційність – використання сучасних технологій для автоматизації та оптимізації процесів (наприклад, систем GPS, IoT, програмного забезпечення для управління логістикою);

У сучасних умовах цифровізації економіки транспортна логістика відіграє вирішальну роль у забезпеченні конкурентоспроможності підприємств. Використання автоматизованих систем управління транспортом (TMS) дозволяє покращити контроль за переміщенням вантажів, автоматизувати планування маршрутів, зменшити ризики помилок і оперативно реагувати на зміни в умовах транспортування.

Важливою складовою транспортної логістики є моніторинг транспортних засобів у режимі реального часу. Це дозволяє мінімізувати затримки, контролювати витрати пального та забезпечувати виконання умов договорів. Інтеграція таких систем із програмним забезпеченням для аналізу даних сприяє прогнозуванню попиту та оптимізації логістичних процесів.

## 1.2 Сучасні підходи до кластеризації даних у логістиці

Кластеризація даних є важливим інструментом в аналізі великих обсягів інформації, що використовується для оптимізації логістичних процесів. Завдяки методам кластеризації можна визначити групи вантажів, маршрутів або клієнтів, які мають схожі характеристики, і таким чином підвищити ефективність управління логістичною системою.

Одним із сучасних підходів є використання алгоритмів машинного навчання, таких як k-means, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) та ієрархічна кластеризація. Наприклад, алгоритм k-means дозволяє розподілити дані на певну кількість кластерів, враховуючи мінімізацію середньоквадратичного відхилення всередині кожного з них. DBSCAN ефективно

працює з нерівномірно розподіленими даними, виявляючи кластери будь-якої форми [4].

Інший підхід базується на географічній кластеризації, яка враховує геопросторові дані для оптимізації маршрутів транспортування. Наприклад, використання алгоритмів НЗ (Hexagonal Hierarchical Geospatial Indexing System) дозволяє адаптувати маршрути до реальних умов, таких як дорожній рух або географічні обмеження.

Крім зазначених методів, у логістиці активно застосовуються алгоритми на основі штучних нейронних мереж, які здатні адаптивно знаходити залежності між характеристиками даних. Наприклад, автоенкодери можуть зменшувати розмірність даних перед кластеризацією, підвищуючи точність поділу.

Методи спектральної кластеризації використовують аналіз власних значень матриць суміжності для виявлення складних структур у даних. Вони ефективні для випадків із великою кількістю параметрів і дозволяють враховувати взаємозв'язки між об'єктами в кластері [5].

Дослідження в цій галузі демонструють, що поєднання алгоритмів кластеризації з прогновною аналітикою дозволяє підприємствам підвищити ефективність логістичних процесів на 30% і знизити витрати на транспортування [4].

На рис. 1.1 показано результати роботи алгоритму k-means, який розподілив точки доставки на 4 кластери, що дозволяє оптимізувати маршрути перевезень.

Одним із перспективних підходів є алгоритм BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies). Він базується на побудові деревоподібної структури даних, як видно з рис. 1.2, що дозволяє обробляти великі обсяги інформації з мінімальними витратами пам'яті. BIRCH ефективний для динамічних даних і використовується в задачах логістики для аналізу великої кількості точок доставки або маршрутів. Основна перевага цього методу – можливість адаптації до потоків даних у реальному часі, що робить його незамінним для моніторингу логістичних процесів [6].

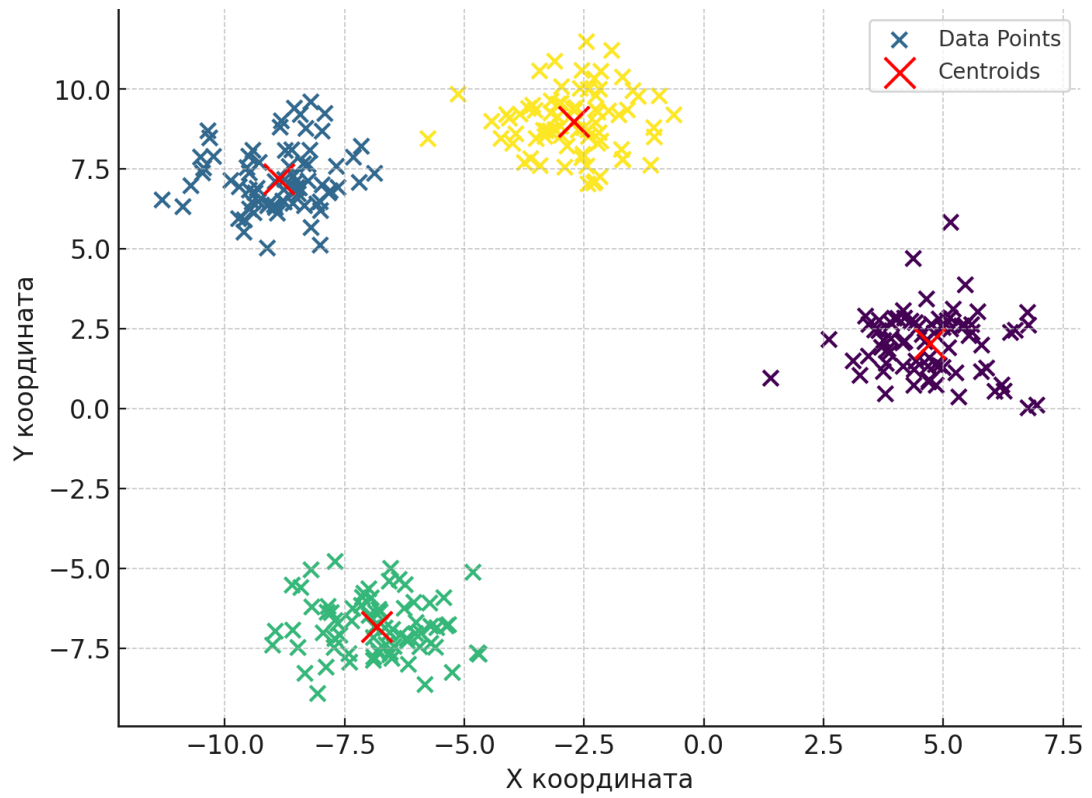


Рисунок 1.1 – Приклад кластеризації даних методом k-means

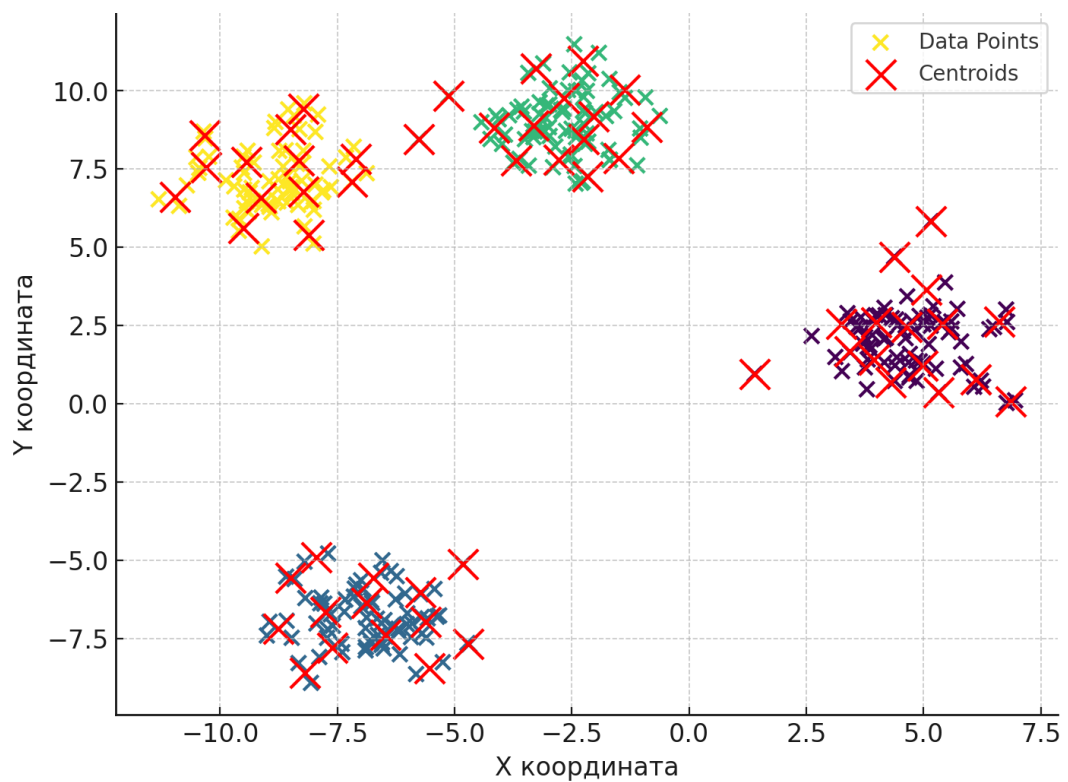


Рисунок 1.2 – Приклад кластеризації даних методом birch

З рис. 1.1-1.2 видно, що K-means більше підходить для симетричних даних із чіткими межами кластерів, тоді як BIRCH краще обробляє нерівномірно розподілені дані й дозволяє створювати кластери різної форми та розміру. Це робить BIRCH корисним для складних задач у логістиці, де дані можуть мати різноманітні структури.

### 1.3 Технології автоматизації логістичних процесів

Автоматизація логістичних процесів є невід'ємною складовою сучасних логістичних систем, яка спрямована на підвищення ефективності управління, зниження витрат і мінімізацію впливу людського фактора. Використання новітніх технологій забезпечує оптимізацію всіх етапів логістичного ланцюга – від закупівлі матеріалів до доставки кінцевому споживачу.

Основними технологіями автоматизації в логістиці є:

– системи управління транспортом (TMS, Transportation Management Systems) Ці системи дозволяють автоматизувати планування маршрутів, моніторинг транспорту в реальному часі та управління витратами на перевезення. Завдяки TMS підприємства можуть скоротити час доставки, зменшити витрати на паливо та покращити обслуговування клієнтів. Наприклад, інтеграція TMS із системами GPS дозволяє відстежувати місцезнаходження транспорту в режимі реального часу;

– системи управління складом (WMS, Warehouse Management Systems), WMS забезпечують автоматизацію операцій на складах, таких як приймання, розміщення, зберігання, комплектація та відвантаження товарів. Ці системи використовують технології штрихового кодування та радіочастотної ідентифікації (RFID, Radio Frequency Identification) для забезпечення точного обліку товарів;

– роботизовані системи, роботизація дозволяє автоматизувати фізичні процеси, такі як переміщення вантажів, комплектація замовлень і пакування. Наприклад, використання автономних мобільних роботів (AMR, Autonomous

Mobile Robots) на складах дозволяє зменшити час виконання операцій і знизити ризики, пов'язані з фізичною працею;

– інтернет речей (IoT, Internet of Things), технології IoT забезпечують інтеграцію різних пристроїв і сенсорів у логістичну систему, що дозволяє отримувати та аналізувати дані в реальному часі. Наприклад, сенсори температури і вологості можуть контролювати умови перевезення чутливих до змін клімату товарів, таких як ліки чи харчові продукти;

– прогнозна аналітика та штучний інтелект (AI, Artificial Intelligence) Штучний інтелект використовується для аналізу великих обсягів даних, прогнозування попиту, оптимізації маршрутів і управління запасами. Наприклад, AI може автоматично коригувати плани доставки, враховуючи реальні умови руху транспорту або зміни попиту. Згідно з даними дослідження, прогнозна аналітика дозволяє підприємствам зменшити витрати на 20% завдяки своєчасній адаптації до змін у попиті [3]. Інтеграція AI із хмарними платформами забезпечує швидкий доступ до результатів аналітики з будь-якого пристрою;

– хмарні технології, хмарні платформи дозволяють забезпечувати доступ до логістичних даних з будь-якого місця, що сприяє підвищенню гнучкості управління. Наприклад, хмарні рішення інтегрують WMS, TMS і IoT для створення єдиної системи управління логістичними процесами. Це забезпечує підприємствам можливість об'єднувати дані з різних джерел у реальному часі та приймати обґрунтовані рішення.

#### 1.4 Аналіз існуючих методів побудови маршрутів та вибір кластеризаційних методів для організації логістичних процесів

Ефективність логістичних процесів значною мірою залежить від оптимізації маршрутів доставки товарів. Це завдання пов'язане зі зниженням витрат, дотриманням термінів доставки та забезпеченням високої якості обслуговування клієнтів. З цією метою використовуються різноманітні алгоритми для побудови маршрутів і кластеризації точок доставки, які мають як переваги, так і обмеження.

Одним із найбільш використовуваних методів є генетичні алгоритми. Цей підхід моделює природний процес еволюції, дозволяючи отримувати оптимальні або близькі до оптимальних рішення навіть у складних умовах. Наприклад, у логістичних мережах з десятками тисяч точок доставки генетичні алгоритми можуть знизити загальні витрати до 18% порівняно з традиційними методами [8].

Мурашині алгоритми (ACO) є ще одним популярним методом. Він базується на поведінці мурах, які шукають найкоротші маршрути між джерелом і ціллю. Цей алгоритм добре працює для задач, де існує багато змінних умов, таких як динамічні дорожні затори чи додаткові точки доставки. Однак його ефективність залежить від правильного налаштування параметрів, таких як рівень випаровування феромону, що моделює зміни в маршрутах.

В останні роки активно розвиваються комбіновані підходи, які інтегрують кілька алгоритмів одночасно. Наприклад, використання генетичних алгоритмів разом із методами прогнозування аналітики дозволяє динамічно змінювати маршрути залежно від змін попиту чи доступності доріг. Це забезпечує високу гнучкість систем і мінімізує ризики затримок.

У логістиці методи кластеризації є основою для групування точок доставки. K-means залишається найбільш популярним завдяки своїй простоті й швидкості. Алгоритм розподіляє точки на задану кількість кластерів, мінімізуючи варіацію всередині кожного з них. Проте його ефективність знижується, якщо дані мають нерівномірну щільність або містять шумові точки.

DBSCAN є методом, який дозволяє формувати кластери на основі щільності. Цей підхід ігнорує шумові дані, що робить його ідеальним для великих наборів точок із нерівномірною щільністю. Наприклад, дослідження [7] показали, що використання DBSCAN у транспортній логістиці дозволяє зменшити кількість пустих рейсів на 20%.

BIRCH використовує деревоподібну структуру для поступового зменшення даних і формування кластерів. Цей метод добре працює з великими наборами даних і може бути застосований у потокових задачах. Наприклад, у логістиці BIRCH

часто використовується для обробки реальних даних у режимі реального часу, дозволяючи динамічно адаптувати маршрути в мегаполісах.

Ієрархічна кластеризація, або агломеративний метод, є ще одним важливим підходом у логістиці. Цей метод передбачає поступове об'єднання точок або кластерів, починаючи з найменшої відстані між ними, доки всі точки не будуть об'єднані в одне ціле. Агломеративний підхід дозволяє побудувати дендрограму – деревоподібну структуру, яка відображає ієрархію кластерів. Це дає змогу аналізувати дані на різних рівнях деталізації. Наприклад, для доставки в різні регіони можна почати з кластеризації на рівні областей, а потім деталізувати до рівня міст чи вулиць.

Агломеративна кластеризація є корисною для задач, де необхідно враховувати багаторівневу структуру даних. Проте вона має свої обмеження: висока обчислювальна складність робить її менш придатною для великих наборів даних без попередньої оптимізації. У логістиці цей метод часто поєднується з іншими підходами, такими як K-means, для початкового розподілу точок.

Для аналізу результатів кластеризації використовуються такі метрики, як коефіцієнт Сілуєта, який показує, наскільки добре точка належить своєму кластеру, і індекс Девіса-Болдіна, що порівнює розміри й відстані між кластерами. Наприклад, високий коефіцієнт Сілуєта свідчить про чітке розмежування між кластерами, що важливо для оптимізації маршрутів. Використання цих метрик у поєднанні з DBSCAN дозволяє досягти адаптивності до змін у реальному часі [9].

Окрім зазначених метрик, у складних логістичних задачах використовуються адаптивні підходи, які поєднують кілька критеріїв. Наприклад, багатокритеріальна оцінка з урахуванням часу доставки, витрат і кількості маршрутів дозволяє зробити більш обґрунтований вибір методу кластеризації. Дослідження [8] продемонстрували, що комбіноване використання індексів Сілуєта та Девіса-Болдіна в задачах доставки дозволяє підвищити точність кластеризації на 18%.

У задачах із чіткими географічними межами, наприклад, доставці в одному районі, найкраще працює K-means. Для задач, де точки розподілені нерівномірно або є багато шумових даних, DBSCAN виявляється значно ефективнішим. У

великих потокових системах, де необхідна динамічна адаптація, BIRCH забезпечує найкращі результати завдяки своїй ітеративній структурі. Для аналізу багаторівневої структури даних агломеративний метод забезпечує глибокий аналіз ієрархій кластерів, але потребує додаткових обчислювальних ресурсів.

Згідно з дослідженнями [8], комбіноване використання кластеризації та прогнозу аналітики може зменшити логістичні витрати до 25%. Успішні приклади включають інтеграцію кластеризації з TMS для автоматичного оновлення маршрутів залежно від умов на дорогах.

Окрім цього, зростає популярність гібридних методів, які поєднують переваги різних підходів. Наприклад, використання DBSCAN для початкового визначення кластерів, а потім застосування BIRCH для оптимізації в реальному часі забезпечує адаптивність і точність навіть у динамічних умовах мегаполісів.

Приклад застосування ансамблевого методу кластеризації для категоріальних даних

У "Використання ансамблю алгоритмів для кластеризації категоріальних даних" описується підхід, що поєднує кілька базових методів кластеризації, таких як K-min та ієрархічна кластеризація, для аналізу категоріальних даних [10]. Цей підхід дозволяє підвищити точність кластеризації завдяки усуненню недоліків окремих алгоритмів та використанню їх сильних сторін у комбінації.

Ансамблевий підхід складається з кількох етапів:

а) початкова кластеризація:

1) на першому етапі до даних застосовуються кілька базових алгоритмів кластеризації, наприклад, K-mins, що добре працює з категоріальними даними;

2) ієрархічна кластеризація додається для створення структурованого уявлення про дані;

б) агрегація результатів:

1) результати базових алгоритмів об'єднуються за допомогою механізму голосування або зважування;

2) для кожного об'єкта даних визначається його кластер відповідно до більшості класифікацій;

в) оптимізація:

1) отримані кластери оцінюються за допомогою метрик, таких як коефіцієнт Сілуета;

2) якщо результат не відповідає очікуванням, модифікуються параметри базових алгоритмів або проводиться додаткова ітерація кластеризації.

Автори статті розглянули дані фізичних осіб, зокрема їх демографічні характеристики: вік, стать, рівень освіти, місце проживання та дохід. Метою дослідження було визначити кластери з найбільш схожими групами людей для подальшого використання в маркетингових дослідженнях.

а) дані:

1) вхідні дані склалися з 10 000 записів, кожен із яких мав 5 категоріальних ознак;

2) дані були попередньо оброблені для усунення пропусків та приведення категорій до уніфікованого формату;

б) кластеризація:

1) алгоритм K-mіns був застосований для поділу даних на 5 попередньо визначених кластерів;

2) ієрархічна кластеризація створила дендрограму, що дозволило переглянути альтернативні варіанти кількості кластерів;

в) результати:

1) у результаті аналізу було виявлено 4 основні кластери:

- молодь із середнім доходом, орієнтована на розваги;
- особи середнього віку з високим доходом, які цікавляться інвестиціями;
- люди пенсійного віку з низьким доходом, які витрачають гроші переважно на медичні послуги;
- сім'ї з дітьми, які спрямовують витрати на освіту та товари для дому;

2) ансамблевий підхід дозволив знизити помилки класифікації на 12% у порівнянні з використанням одного K-mіns;

г) висновки:

1) використання ансамблевого методу підвищило стійкість результатів кластеризації;

2) підхід дозволив ідентифікувати групи з чітко визначеними характеристиками, що важливо для сегментації ринку та цільового маркетингу.

Переваги ансамблевого підходу:

– підвищення точності кластеризації за рахунок об'єднання кількох моделей;

– можливість роботи з великими наборами категоріальних даних;

– гнучкість у налаштуванні та оптимізації параметрів.

## 1.5 Аналіз програмного забезпечення для моделювання логістичних систем

Моделювання логістичних систем потребує використання спеціалізованого програмного забезпечення, яке забезпечує високий рівень гнучкості, потужності та зручності у використанні. Для цього застосовуються різноманітні програмні мови, серед яких найпоширенішими є Python, Java, C++, R та MATLAB. Кожна з цих мов має унікальні можливості, що визначають їх придатність для вирішення специфічних задач. Основною мовою, яка буде використана у розробці цієї логістичної системи, є Python, завдяки його універсальності та багатофункціональності.

Python вважається однією з найкращих мов програмування для задач моделювання та аналізу логістичних процесів. Його популярність пояснюється простим синтаксисом, що робить його доступним навіть для початківців, і наявністю великої кількості бібліотек для обробки даних, візуалізації та машинного навчання. Згідно з дослідженням [11], Python забезпечує інтеграцію з сучасними інструментами аналізу даних, такими як SQL, Hadoop і Spark, що є ключовим у великих логістичних проектах.

Бібліотека NumPy забезпечує роботу з багатовимірними масивами та виконання числових обчислень, що є основою для моделювання складних процесів.

Вона використовується для операцій з великими наборами даних і підтримує інтеграцію з іншими бібліотеками. Наприклад, функції NumPy дозволяють швидко виконувати лінійну алгебру, сортування та маніпуляції з матрицями, що є основою для оптимізації маршрутів у логістиці[12].

Pandas – це бібліотека, яка забезпечує роботу з табличними даними. Вона дозволяє швидко обробляти великі масиви інформації, виконувати фільтрацію, агрегацію та обчислення за групами даних. У задачах логістики Pandas допомагає аналізувати дані про маршрути, час доставки та інші важливі показники, спрощуючи прийняття рішень[13].

Бібліотека SciPy розширює можливості NumPy і пропонує інструменти для оптимізації, інтеграції, статистичного аналізу та розв'язання диференціальних рівнянь. У задачах логістики це дозволяє виконувати точні математичні розрахунки для побудови оптимальних маршрутів або аналізу ефективності логістичних процесів. Matplotlib є потужним інструментом для створення графіків і візуалізації даних, що допомагає представляти результати моделювання в наочному вигляді.

Для роботи з базами даних у Python часто використовується MySQL. MySQL – це система управління реляційними базами даних, яка дозволяє ефективно працювати з великими наборами даних. Інструмент MySQL Workbench забезпечує візуальний інтерфейс для управління базами даних, їх моделювання, створення запитів і адміністрування. У логістиці MySQL Workbench застосовується для зберігання даних про товари, маршрути, клієнтів та інші критичні елементи системи. Взаємодія між Python і MySQL реалізується за допомогою бібліотек, таких як MySQL Connector або SQLAlchemy, які дозволяють створювати запити безпосередньо з коду[14], [15].

Крім того, для роботи з географічними даними Python підтримує формат GeoJSON. GeoJSON є форматом, який використовується для кодування географічних структур, таких як точки, лінії та полігони. У логістичних системах GeoJSON використовується для зберігання даних про маршрути, склади та зони доставки. У PyCharm можливе інтегрування бібліотек для роботи з GeoJSON, таких

як Fiona або GeoPandas, що забезпечує обробку геопросторових даних безпосередньо в кодї.

PyCharm, інтегроване середовище розробки (IDE, Integrated Development Environment), є ще однією причиною популярності Python серед розробників логістичних систем. PyCharm підтримує автоматичне завершення коду, виявлення помилок, інтеграцію з Git та можливість роботи з віртуальними середовищами. Це дозволяє значно скоротити час на написання та тестування коду, забезпечуючи зручність і ефективність розробки. Крім того, PyCharm має вбудовану підтримку роботи з фреймворками для машинного навчання, такими як TensorFlow та PyTorch, що є особливо корисним у логістичних задачах, де необхідна обробка великих обсягів даних.

Python має величезну спільноту розробників, які постійно створюють нові бібліотеки та розширення. Наприклад, бібліотека Seaborn розширює можливості Matplotlib для створення більш естетичних і складних графіків, що важливо для презентацій результатів моделювання. Інструмент Plotly дозволяє створювати інтерактивні графіки, що особливо корисно для аналізу даних у реальному часі.

Крім Python, у сфері логістики також застосовуються інші мови програмування. Наприклад, Java відома своєю стабільністю і використовується для створення масштабованих корпоративних систем. C++ забезпечує високу продуктивність і підходить для розробки ресурсомістких програм. MATLAB є ефективним інструментом для виконання математичних і технічних розрахунків, але його обмежена інтеграція з іншими системами зменшує гнучкість у задачах логістики.

Таким чином, Python виявився найкращим вибором для реалізації цієї логістичної системи завдяки його простоті, широким можливостям, доступу до безкоштовних інструментів і підтримці великої спільноти. Інструменти, такі як PyCharm, і бібліотеки, такі як NumPy, Pandas, SciPy та Matplotlib, забезпечують потужний набір функцій для розробки, тестування та оптимізації програмного забезпечення.

## 1.6 Висновки до першого розділу

У першому розділі було проведено комплексний аналіз теоретичних основ і сучасного стану розробок у сфері логістики. Логістика, як одна з ключових галузей сучасної економіки, потребує постійного вдосконалення методів і підходів до управління процесами доставки товарів, оптимізації маршрутів і використання ресурсів. У результаті дослідження було визначено, що застосування сучасних технологій, зокрема методів кластеризації та програмного забезпечення, є основою для ефективного функціонування логістичних систем.

У підрозділі, присвяченому основам транспортної логістики, було розглянуто ключові принципи та завдання, які забезпечують ефективне функціонування транспортних систем. Основні акценти було зроблено на раціоналізації маршрутів, гнучкості в управлінні перевезеннями та використанні сучасних інформаційних технологій для підвищення ефективності. Транспортна логістика є базовою складовою загальної логістичної системи, що визначає її важливість у процесах планування та організації.

Дослідження методів кластеризації даних показало, що сучасні алгоритми, такі як K-means, DBSCAN, BIRCH та агломеративна кластеризація, мають широкий спектр застосувань у логістиці. Вибір конкретного методу залежить від специфіки даних та поставлених завдань. Було також визначено, що використання відповідних метрик для оцінки результатів кластеризації, таких як коефіцієнт Сілуета та індекс Девіса-Болдіна, дозволяє підвищити точність аналізу та оптимізації логістичних процесів.

Особливу увагу приділено аналізу програмного забезпечення для моделювання логістичних систем. Python, завдяки своїй універсальності, простоті та широкому спектру бібліотек, було визначено як оптимальний інструмент для розробки програмних рішень у цій галузі. Використання бібліотек, таких як NumPy, Pandas, SciPy та Matplotlib, забезпечує високу продуктивність і точність обробки даних. Інтегроване середовище розробки PyCharm сприяє ефективній реалізації

програмних проектів, забезпечуючи підтримку інструментів для машинного навчання та роботи з базами даних.

Також було розглянуто використання реляційних баз даних на прикладі MySQL та MySQL Workbench, які забезпечують зручність у роботі з великими масивами інформації. Інтеграція Python із GeoJSON дозволяє ефективно працювати з географічними даними, що є критично важливим для задач оптимізації маршрутів і зонування в логістиці. Завдяки таким інструментам логістичні системи стають більш гнучкими, ефективними та адаптивними до змін зовнішнього середовища.

Проведений аналіз підтверджує, що сучасні інформаційні технології, інтеграція даних та аналітичні інструменти є основою для розвитку логістики в умовах цифрової трансформації. Впровадження автоматизації, використання методів машинного навчання та прогнозової аналітики дозволяє підприємствам не лише оптимізувати логістичні процеси, але й забезпечити високий рівень обслуговування клієнтів.

Таким чином, висновки першого розділу підтверджують необхідність комплексного підходу до розробки логістичних систем, який включає аналіз теоретичних основ, вибір оптимальних методів кластеризації, використання сучасного програмного забезпечення та інтеграцію новітніх технологій для вирішення практичних завдань у сфері логістики.

## 2 ПРОЕКТУВАННЯ ЛОГІСТИЧНОЇ СИСТЕМИ НА ВИРОБНИЦТВІ

### 2.1 Постановка задачі

Сучасна транспортна логістика стикається з численними проблемами, серед яких необхідність оптимізації маршрутів, зниження витрат на перевезення, підвищення ефективності процесів управління та інтеграція цифрових технологій. Постійна мінливість умов ринку, глобалізація логістичних ланцюгів та потреба у швидкому реагуванні на зовнішні фактори вимагають впровадження рішень, які дозволять адаптуватися до складної динаміки. Важливе місце посідає автоматизація, що зменшує людський фактор та прискорює прийняття рішень.

Основна задача, яка ставиться у даному підрозділі, полягає у створенні логістичної системи, яка буде:

- забезпечувати оптимальне планування маршрутів доставки вантажів на підприємстві;
- мінімізувати витрати на транспортування, включаючи фінансові та часові ресурси;
- інтегрувати сучасні технології автоматичного управління для аналізу, кластеризації та прогнозування логістичних процесів;
- забезпечувати стійкість та гнучкість системи в умовах динамічних змін зовнішніх факторів, таких як економічна нестабільність, сезонні коливання або раптові кризи.

Для реалізації поставлених цілей буде проведено глибокий аналіз наявних даних про вантажоперевезення з їх подальшою кластеризацією за географічними та іншими ключовими ознаками. Це стане основою для ефективного групування замовлень і визначення оптимальних маршрутів. Будуть використані алгоритми оптимізації на основі теорії графів, які дозволять побудувати найкоротші маршрути з урахуванням обмежень, таких як вага вантажу, габарити, графік доставки тощо.

Ключовим елементом буде використання методів машинного навчання для прогнозування можливих змін у логістичних процесах. Це дозволить не лише

швидко реагувати на зміни, але й передбачати можливі ризики. Програмне забезпечення, яке буде розроблено, виконуватиме функції автоматичного моделювання маршрутів, аналізу ефективності логістичних операцій та візуалізації отриманих даних.

Очікуваними результатами впровадження цієї системи є суттєве скорочення часу доставки, зниження витрат на логістику, підвищення якості обслуговування клієнтів та збільшення прозорості процесів управління. Усе це буде сприяти підвищенню конкурентоспроможності підприємства на ринку.

Таким чином, визначено шляхи вирішення поставленої задачі та окреслено ключові вимоги до майбутньої логістичної системи. Подальші етапи роботи будуть спрямовані на деталізацію алгоритмів та їх інтеграцію в програмне забезпечення, що розробляється.

## 2.2 Розробка алгоритму побудови оптимальних маршрутів на виробництві

Основна мета підрозділу – визначення чіткої послідовності дій для створення алгоритму, який відповідає потребам сучасної транспортної логістики. Процес розробки базується на аналізі та вивченню зібраних даних предметної області, спрямований на інтеграцію всіх необхідних компонентів у єдину систему.

Ефективна побудова маршрутів є ключовою задачею логістичної системи, оскільки вона дозволяє мінімізувати витрати, зменшити час доставки та підвищити якість обслуговування клієнтів. Для цього необхідно враховувати такі аспекти, як характеристики вантажів, можливості транспортних засобів, географічні особливості маршрутів і часові обмеження. Розроблений алгоритм має забезпечувати гнучкість та адаптивність до змінних умов.

Алгоритм побудови оптимальних маршрутів:

а) збір і підготовка даних:

1) реалізувати збір даних про замовлення: адреси доставки, характеристики вантажів, терміни виконання;

2) зібрати інформацію про транспортні засоби, включаючи їх вантажопідйомність, поточне місце розташування та доступність;

3) визначити географічні координати точок доставки;

4) очистити та обробити дані для усунення помилок і дублікатів;

б) кластеризація точок доставки:

1) реалізувати кластеризацію точок доставки на основі географічного принципу, використовуючи алгоритми, закладені у програму;

2) визначити оптимальну кількість кластерів, враховуючи транспортні обмеження та відстані між точками;

3) перевірити результати кластеризації та створити початкові групи для побудови маршрутів;

в) формування початкових маршрутів:

1) розробити базові маршрути, враховуючи мінімізацію відстаней між точками та обмеження транспортних засобів;

2) впровадити сортування точок доставки в межах кожного кластера;

3) перевірити відповідність маршрутів обмеженням щодо ваги та габаритів вантажів;

г) оптимізація маршрутів:

1) використати методи оптимізації для мінімізації витрат або часу доставки;

2) інтегрувати перестановки точок у маршрутах для покращення їх ефективності;

3) зберегти оптимальні маршрути у окремому файлі;

г) прогнозування змін і адаптація:

1) інтегрувати механізми прогнозування змін, таких як зміни адреси доставки, вплив географічних об'єктів та інші фактори;

2) реалізувати автоматичне коригування маршрутів у реальному часі;

д) візуалізація та тестування:

1) забезпечити візуалізацію маршрутів на інтерактивній карті для зручності перевірки;

- 2) провести тестування алгоритму на реальних і змодельованих даних;
- 3) оцінити ефективність маршрутів за ключовими показниками, такими як відстань і кількість маршрутів;

е) інтеграція в систему:

- 1) впровадити алгоритм у логістичну систему з підтримкою автоматизованого управління;
- 2) розробити інтерфейс користувача для перегляду, редагування та експорту маршрутів.

Реалізація алгоритму базується на використанні сучасних технологій і програмного забезпечення, що забезпечують комплексність і точність виконання завдань. Для програмування використовується мова Python, яка забезпечує гнучкість у написанні алгоритмів та інтеграції додаткових модулів. Застосовуються такі бібліотеки, як:

- scikit-learn для кластеризації та роботи з алгоритмами машинного навчання;
- Google Maps API для отримання географічних координат, побудови маршрутів та інтеграції картографічних сервісів;
- Folium для візуалізації маршрутів на інтерактивних картах, що дозволяє наочно представити результати роботи алгоритму;
- NumPy та SciPy для виконання математичних обчислень та оптимізації маршрутів.

Інформація про замовлення, транспортні засоби та географічні координати точок доставки зберігається у базі даних MySQL. Цей вибір бази даних обумовлений її високою швидкістю обробки запитів та можливістю інтеграції з Python через бібліотеку mysql-connector-python. Інформація про згенеровані кластери та візуалізація отриманих результатів зберігається у файлах.

Розробка алгоритму передбачає також інтеграцію з інструментами автоматизації процесів. Наприклад, використання pickle для серіалізації проміжних даних дозволяє прискорити обробку інформації. Для обробки великих обсягів

даних передбачено використання багатопоточності, що реалізується через бібліотеку `concurrent.futures`.

Тестування алгоритму проводиться з використанням реальних і змодельованих даних. Для оцінки ефективності використовуються метрики оцінки кластеризації. Результати тестування аналізуються для виявлення можливостей оптимізації та покращення точності алгоритму.

Упровадження алгоритму в логістичну систему включає розробку зручного інтерфейсу користувача, який дозволяє переглядати, редагувати та експортувати маршрути. Для цього використовується бібліотека Tkinter або інші інструменти створення графічних інтерфейсів, залежно від вимог користувача.

Запропонований алгоритм побудови оптимальних маршрутів дозволяє врахувати всі аспекти логістичного процесу та інтегрувати їх у програмне забезпечення. Використання сучасних методів і технологій забезпечує високу точність, ефективність та гнучкість системи. Це сприятиме досягненню високого рівня обслуговування клієнтів і підвищенню конкурентоспроможності підприємства на ринку.

### 2.3 Інтеграція кластеризаційних даних в логістичну систему

Ключовим етапом для підвищення ефективності побудови маршрутів доставки є групування точок доставки за різними критеріями, такими як географічне розташування, характеристики вантажів або часові обмеження. Для досягнення цієї мети в програмі використовуються п'ять основних алгоритмів кластеризації: `k-means`, `DBSCAN`, `Agglomerative`, `Spectral` та `Birch` [16]-[18]. Кожен з цих методів має свої особливості і адаптований для вирішення конкретних завдань логістичної системи, що забезпечує їх гнучку інтеграцію.

Методи кластеризації інтегруються у програму через модульну архітектуру, яка дозволяє гнучко обирати алгоритм залежно від специфіки задачі. Алгоритми отримують доступ до єдиного джерела даних, яке містить географічні координати точок доставки, параметри вантажів, часові обмеження та типи об'єктів доставки.

Для забезпечення зручності користувачів у програмі реалізовано можливість ручного налаштування параметрів кластеризації через графічний інтерфейс користувача (GUI). Цей підхід дає змогу змінювати кількість кластерів, поріг щільності або інші важливі параметри, залежно від вимог задачі. Модульний підхід у кодї забезпечує легке додавання нових алгоритмів кластеризації або модифікацію існуючих методів, що розширює функціональність програми.

Дані для кластеризації включають такі ключові параметри, як географічні координати точок доставки, характеристики вантажів (вага, об'єм, специфічні умови транспортування). Для підвищення точності роботи алгоритмів у програмі реалізовано фільтрацію точок за їх характеристиками. Наприклад, точки, здатні приймати важкі чи великогабаритні вантажі, обробляються окремо, з урахуванням наявності рампи або іншого спеціалізованого обладнання. Інші точки, що мають обмеження по вазі чи розмірах вантажів, кластеризуються за окремими параметрами, що дозволяє оптимізувати маршрути для різних типів вантажів

Метод k-means є одним із найпоширеніших алгоритмів кластеризації завдяки його простоті та швидкості. Він розподіляє точки на задану кількість кластерів, мінімізуючи відстань між точками й центроїдами кластерів. Основними вхідними параметрами є кількість кластерів та координати точок. Алгоритм працює ітеративно, оновлюючи положення центроїдів кластерів до тих пір, поки вони не стабілізуються. Для оцінки результатів використовується метрика `silhouette_score`, яка дозволяє визначити, наскільки добре точки належать до своїх кластерів і чи є вони чітко відокремленими від інших кластерів. Цей метод ідеально підходить для рівномірно розподілених даних, проте менш ефективний для точок із різною щільністю або з шумами.

Метод DBSCAN базується на підході кластеризації на основі щільності, що дозволяє виділяти кластери довільної форми та ігнорувати шумові точки. Вхідними параметрами є  $\epsilon$  (радіус) та мінімальна кількість точок у кластері. Перевагою DBSCAN є його здатність працювати з даними, які мають нерівномірну щільність, наприклад, у складних міських умовах. Він ефективно виділяє густонаселені області та відокремлює їх від малонаселених. Недоліком є те, що вибір параметрів

є та мінімальної кількості точок значно впливає на результати, тому ці параметри потребують ретельного налаштування через тестування на реальних даних.

Метод *Agglomerative* є ієрархічним підходом до кластеризації, який поступово об'єднує точки або кластери на основі певної міри відстані, наприклад, евклідової. Цей метод є особливо корисним для структурованих даних, таких як точки у сільській місцевості, де відстань між точками може бути значною. Результатом роботи є дендрограма, яка візуалізує ієрархічну структуру об'єднань. Користувач може самостійно визначити рівень, на якому слід зупинити об'єднання для формування оптимальних кластерів.

Метод *Spectral* кластеризації використовує спектральні властивості матриці подібності між точками для виявлення кластерів. Він ефективний для даних, які мають складні зв'язки, наприклад, коли важливо враховувати як географічну близькість, так і часові характеристики точок доставки. Алгоритм спочатку зменшує розмірність даних за допомогою спектрального розкладу, а потім застосовує стандартні методи кластеризації, такі як *k-means*. Перевагою цього підходу є його гнучкість у роботі з різними типами даних, проте він потребує значних обчислювальних ресурсів.

Метод *Birch* оптимізований для роботи з великими масивами даних. Він використовує деревоподібну структуру для поступового групування точок, що дозволяє ефективно обробляти тисячі точок доставки. *Birch* виділяється своєю здатністю адаптуватися до змін у даних і підтримувати кластеризацію в реальному часі. Цей метод забезпечує швидке групування навіть за наявності великих обсягів даних, проте може бути менш точним у роботі з нерівномірно розподіленими точками.

Усі зазначені методи кластеризації інтегровані у програму через модульну архітектуру, що дозволяє обирати оптимальний підхід залежно від характеру даних та поставлених завдань. Їх налаштування проводиться через графічний інтерфейс користувача, де користувач може змінювати параметри, такі як кількість кластерів у *k-means* або радіус у *DBSCAN*. Це забезпечує гнучкість і високу адаптивність системи до умов конкретних логістичних операцій.

Навчання кластеризаційної моделі базується на багаторазовому тестуванні кожного алгоритму з різними наборами даних, що дозволяє гнучко адаптувати модель до конкретних задач. У програмному застосунку реалізовано автоматизовану послідовність запуску алгоритмів кластеризації з різними параметрами. Для кожного набору даних проводиться аналіз за кількома ключовими критеріями, такими як кількість точок у кластері, географічна близькість точок до географічних об'єктів, особливості вантажу (вага, об'єм). Наприклад, точки, що приймають великогабаритні вантажі та обладнані рампами, кластеризуються окремо від точок, які мають обмеження на прийом великогабаритних вантажів. Такий підхід забезпечує оптимальне розподілення точок залежно від їх характеристик.

Для кожного алгоритму передбачено ретельну перевірку параметрів. Наприклад, у методі DBSCAN налаштовується параметр  $\epsilon$  (радіус сусідства) та мінімальна кількість точок у кластері, що дозволяє обирати оптимальні значення залежно від щільності точок у даних. У k-means проводиться варіювання кількості кластерів для досягнення найкращого розподілу точок. Результати тестування кожного алгоритму зберігаються у окремому файлі, що дозволяє створити бібліотеку налаштувань для різних сценаріїв і швидко застосовувати їх у майбутніх задачах.

Результати кластеризації відображаються на інтерактивній карті, створеній за допомогою бібліотеки Folium. Кожен кластер позначається унікальним кольором, що спрощує візуалізацію розподілу точок. Інтерактивність карти дозволяє користувачеві деталізувати інформацію про кожну точку, включаючи її географічне розташування та характеристику вантажу. У разі виявлення невідповідностей користувач може коригувати параметри алгоритму безпосередньо через графічний інтерфейс програми.

У програмі також реалізовано функцію порівняння результатів різних алгоритмів на одному наборі даних. Це дозволяє визначити, який алгоритм забезпечує найкращу якість кластеризації для конкретної задачі. Для оцінки результатів використовуються метрики `silhouette_score`, `calinski_harabasz_score` та

davies\_bouldin\_score. Наприклад, silhouette\_score допомагає оцінити чіткість розподілу точок у кластерах, тоді як calinski\_harabasz\_score дозволяє визначити ступінь роздільності кластерів.

Ще однією важливою функцією є адаптивне навчання моделі. Після кожного циклу кластеризації система аналізує отримані результати та автоматично пропонує нові налаштування параметрів для подальшого покращення. Наприклад, якщо silhouette\_score вказує на слабку роздільність кластерів, система може запропонувати змінити кількість кластерів або поріг щільності у DBSCAN. Цей процес дозволяє користувачеві не лише працювати з готовими моделями, але й активно брати участь у їх навчанні та оптимізації.

Загалом, інтеграція навчання моделі та відображення результатів у програмному застосунку забезпечує високу ефективність роботи з кластеризаційними даними. Завдяки гнучкому налаштуванню алгоритмів, візуалізації результатів та адаптивному підходу до навчання, система відповідає потребам сучасних логістичних задач і дозволяє швидко адаптуватися до змінних умов.

## 2.4 Використання методів машинного навчання для оптимізації логістичних процесів

Сучасні технології дозволяють значно підвищити ефективність логістичних операцій завдяки аналізу великих обсягів даних, прогнозуванню попиту, оптимізації маршрутів і автоматизації рішень. У цьому підрозділі викладено основні підходи, що базуються на машинному навчанні, та їх інтеграцію в логістичну систему підприємства.

Методи машинного навчання в логістиці включають як супервізовані, так і несупервізовані підходи. Супервізовані методи, такі як лінійна регресія, дерева рішень і градієнтний бустинг, використовуються для прогнозування попиту, часу доставки або витрат. Наприклад, методи регресії дозволяють передбачити обсяг перевезень на основі історичних даних про замовлення, сезонних коливань та

інших параметрів. Це забезпечує точніше планування ресурсів і запобігання перенавантаженням системи [19].

Несупервізовані методи, такі як кластеризація та алгоритми зменшення розмірності, використовуються для групування клієнтів або точок доставки за схожими характеристиками. Наприклад, алгоритми кластеризації, такі як k-means або DBSCAN, допомагають групувати точки доставки залежно від їх географічного розташування або характеристик вантажу. Це спрощує планування маршрутів і знижує витрати на транспортування [20].

У програмному застосунку методи машинного навчання інтегруються через модульну архітектуру. Для прогнозування попиту використовується історичний аналіз замовлень, що дозволяє враховувати сезонні зміни, поведінку клієнтів і зовнішні фактори, такі як погодні умови. Наприклад, рекурентні нейронні мережі (RNN) застосовуються для аналізу часових рядів, що дозволяє передбачати майбутні замовлення на основі попередніх даних [21].

Оптимізація маршрутів здійснюється через комбінацію алгоритмів машинного навчання та традиційних методів оптимізації, таких як методи гілок і меж або генетичні алгоритми. Машинне навчання використовується для оцінки можливих сценаріїв і вибору найбільш ефективного варіанту. Наприклад, градієнтний бустинг використовується для моделювання залежностей між характеристиками маршрутів і їх ефективністю [22]

Для автоматизації прийняття рішень у логістичних процесах використовуються алгоритми багатокритеріальної оптимізації. Вони враховують такі фактори, як час доставки, витрати на паливо, екологічні обмеження та пріоритети клієнтів. Наприклад, методи багатокритеріального аналізу дозволяють обрати оптимальний маршрут, що мінімізує витрати та забезпечує дотримання дедлайнів [22].

## 2.5 Застосування теорії автоматичного управління

Для оптимізації процесів у логістичній системі визначимо передавальну функцію системи доставки вантажів і виконаємо аналіз її стійкості.

Передавальна функція описує залежність між вхідними і вихідними параметрами системи в операторній формі [26]. Вхідними параметрами є кількість замовлень  $Q$ , які надходять у систему, а вихідними – час доставки  $T$ . Розглянемо спрощену модель логістичної системи, де час доставки залежить від кількості замовлень і швидкості обробки кожного замовлення:

$$T(s) = \frac{K}{\tau s + 1} Q(s), \quad (2.1)$$

де:  $T(s)$  – вихідний сигнал (час доставки);

$Q(s)$  – вхідний сигнал (кількість замовлень);

$K$  – коефіцієнт пропорційності, що відображає швидкість обробки замовлень;

$\tau$  – стала часу, яка визначає інерційність системи;

$s$  – оператор Лапласа.

Розрахунок параметрів системи:

для проведення аналізу використовуємо наступні експериментальні дані:

– середня кількість замовлень за годину:  $Q_{avg} = 100$ ;

– середній час доставки:  $T_{avg} = 2$  години;

– максимальна продуктивність системи:  $P_{max} = 150$  замовлень/год.

коефіцієнт пропорційності  $K$ :

$$K = \frac{T_{avg}}{Q_{avg}} = \frac{150}{100} = 1.5,$$

стала часу  $\tau$ :

$$\tau = \frac{T_{avg}}{K} = \frac{2}{1.5} \approx 1.33 \text{ години.}$$

Аналіз стійкості системи:

Для перевірки стійкості системи використовуємо характеристичне рівняння передавальної функції:

$$\tau s + 1 = 0,$$

корінь рівняння:

$$s = -\frac{1}{\tau} = -\frac{1}{1.33} \approx -0.75.$$

Оскільки корінь характеристичного рівняння знаходиться у лівій напівплощині комплексної площини, система є стійкою.

Висновок:

Отримана передавальна функція логістичної системи дозволяє описати залежність між кількістю замовлень і часом доставки. Проведений аналіз стійкості показав, що система є стійкою за заданих параметрів. Розрахунок параметрів  $K$  і  $\tau$  може бути використаний для налаштування системи, щоб мінімізувати час доставки і забезпечити ефективне функціонування логістичних процесів.

## 2.6 Висновки до другого розділу

У другому розділі було проведено детальний аналіз та розробку ключових аспектів побудови логістичної системи для оптимізації маршрутів і управління вантажоперевезеннями. У підрозділах розглянуто постановку задачі, алгоритмічні рішення для кластеризації, методи машинного навчання, а також застосування теорії автоматичного управління для покращення ефективності логістичних процесів.

На етапі постановки задачі було визначено основні вимоги до логістичної системи, серед яких оптимізація маршрутів, зниження витрат і забезпечення

дотримання часових обмежень доставки. Для вирішення цих задач запропоновано використання сучасних методів кластеризації, машинного навчання та елементів автоматичного управління.

Алгоритми кластеризації, такі як k-means, DBSCAN, Agglomerative, Spectral та Birch, були інтегровані в систему для групування точок доставки залежно від географічних координат, характеристик вантажів і часових обмежень. Це дозволило покращити планування маршрутів і забезпечити гнучкість у виборі оптимального підходу для кожної ситуації.

Застосування методів машинного навчання, зокрема супервізованих і несупервізованих підходів, сприяло автоматизації логістичних процесів, прогнозуванню попиту і підвищенню точності планування.

Теорія автоматичного управління була використана для моделювання логістичної системи за допомогою передавальних функцій. Проведені розрахунки параметрів системи і аналіз її стійкості показали, що розроблена модель є ефективною для забезпечення стабільної роботи в умовах реальних операцій. Отримані результати демонструють можливість покращення роботи системи за рахунок налаштування параметрів, таких як коефіцієнт пропорційності та стала часу.

Використані підходи та інструменти забезпечують інтеграцію сучасних технологій і теоретичних методів у практичну реалізацію, спрямовану на досягнення високої ефективності логістичних процесів.

## 3 РОЗРОБЛЕННЯ ПРОГРАМИ

### 3.1 Застосування методів кластеризації

Методи кластеризації відіграють важливу роль у групуванні населених пунктів для оптимізації логістичних процесів. Їх застосування дозволяє зменшити витрати на транспортування, покращити якість маршрутизації та ефективно використовувати ресурси. У цьому підпункті розглядається, як основні алгоритми кластеризації інтегровані у програму, а також пояснюється їхня роль у логістичній системі.

Першим алгоритмом, що використовується, є k-means. Він розподіляє точки на задану кількість кластерів, мінімізуючи відстань між точками і центроїдами. У програмі цей алгоритм викликається за допомогою функції:

```
labels, centers = apply_kmeans(data, n_clusters=5)
```

Тут задається кількість кластерів, а результатом є мітки кластерів для кожної точки та їхні центри, що використовуються для візуалізації.

Другий алгоритм – DBSCAN, який виділяє кластери довільної форми і враховує шуми в даних. Його застосування:

```
labels = apply_dbscan(data, eps=0.5, min_samples=5)
```

Параметри `eps` та `min_samples` дозволяють налаштувати алгоритм залежно від щільності точок, що робить його ефективним для нерівномірних розподілів.

Agglomerative кластеризація виконує ієрархічне об'єднання точок. Її виклик у програмі виглядає так:

```
labels = apply_agglomerative(data, n_clusters=4)
```

Цей метод корисний для побудови дендрограм, які візуалізують ієрархічні зв'язки між точками.

Метод Spectral кластеризації, що враховує спектральні властивості матриці подібності, ефективний для складних зв'язків між точками. Його застосування:

```
labels = apply_spectral(data, n_clusters=3)
```

Spectral кластеризація дозволяє враховувати як географічні, так і додаткові параметри, такі як часові обмеження чи тип вантажу.

Останній розглянутий метод – Birch, оптимізований для великих наборів даних. Його виклик у програмі:

```
labels = apply_birch(data, n_clusters=6)
```

Цей алгоритм забезпечує швидке групування навіть за наявності тисяч точок, використовуючи деревоподібну структуру для поступового об'єднання точок.

Дані про населені пункти включають їхні географічні координати, кількість населення та інші характеристики. Наприклад, координати використовуються для розрахунку відстаней між пунктами, що допомагає оптимізувати маршрути. Алгоритми кластеризації дозволяють автоматизувати процеси групування, зокрема розподіл точок доставки залежно від їхньої близькості або інших параметрів.

Результати кластеризації візуалізуються на інтерактивній карті за допомогою бібліотеки Folium. Ця бібліотека забезпечує зручний інструментарій для створення географічних карт, які можна інтерактивно змінювати. Кластери позначаються різними кольорами, що дозволяє легко аналізувати результати. Зокрема, для створення маркерів на карті використовується наступний код:

```
folium.CircleMarker(
    location=(city[2], city[3]),
    radius=13,
    color="black",
    fill=True,
    fill_color=color,
    fill_opacity=0.9,
    weight=0.5,
    popup=f"Cluster: {cluster_id + 1}"
).add_to(map_clusters)
```

Цей приклад додає круговий маркер для кожного населеного пункту, відображаючи кластер, до якого він належить. Параметр popup дозволяє показувати додаткову інформацію, наприклад номер кластера.

У програмі реалізовано підхід до навчання моделей кластеризації з використанням сіткового пошуку параметрів. Налаштування параметрів кожного алгоритму кластеризації проводиться за допомогою словника параметрів `param_grid`:

```
param_grid = {
    'kmeans': {
        'n_clusters': range(2, 16),
        'n_init': [10, 20],
        'random_state': [42]
    },
    'dbscan': {
        'eps': np.arange(0.2, 1.6, 0.1),
        'min_samples': range(2, 6)
    },
    'agglomerative': {
        'n_clusters': range(2, 16)
    },
    'spectral': {
        'n_clusters': range(2, 16),
        'random_state': [42],
        'affinity': ['nearest_neighbors', 'rbf']
    },
    'birch': {
        'n_clusters': range(2, 16)
    }
}
```

Для кожної комбінації параметрів виконується ініціалізація та навчання відповідного алгоритму:

```
for param_comb in itertools.product(*param_grid[algorithm].values()):
    param_dict = dict(zip(param_grid[algorithm].keys(), param_comb))
```

```

if algorithm == 'kmeans':
    model = KMeans(**param_dict)
elif algorithm == 'dbscan':
    model = DBSCAN(**param_dict)
elif algorithm == 'agglomerative':
    model = AgglomerativeClustering(**param_dict)
elif algorithm == 'spectral':
    model = SpectralClustering(**param_dict)
elif algorithm == 'birch':
    model = Birch(**param_dict)
clusters = model.fit_predict(numeric_data)

```

Результати навчання аналізуються на основі метрик якості кластеризації, таких як `silhouette_score`, що дозволяє оцінити чіткість розподілу точок між кластерами. Такий підхід гарантує вибір оптимальних параметрів для кожного алгоритму залежно від даних.

Після завершення розрахунків результати кластеризації зберігаються у два текстові файли: один для точок із рампою, інший – без неї. Код для запису:

```

with open("cluster_results_with_ramp.txt", "w") as file_with_ramp, \
    open("cluster_results_without_ramp.txt", "w") as file_without_ramp:
    for point, cluster_id in zip(data_with_ramp, clusters_with_ramp):
        file_with_ramp.write(f"Point: {point}, Cluster: {cluster_id}\n")
    for point, cluster_id in zip(data_without_ramp, clusters_without_ramp):
        file_without_ramp.write(f"Point: {point}, Cluster: {cluster_id}\n")

```

Цей підхід дозволяє зберігати результати кластеризації в окремі файли, забезпечуючи їх подальше використання для аналізу або візуалізації. Запис у різні файли дає змогу легко розрізняти точки залежно від їхніх характеристик

### 3.2 Візуалізація отриманих даних

Візуалізація є важливим етапом обробки результатів кластеризації, оскільки вона дозволяє краще зрозуміти структуру даних і полегшує аналіз. У цьому підпункті розглядається, як результати кластеризації візуалізуються за допомогою бібліотеки Folium, а також пояснюються основні рядки коду, які відповідають за створення інтерактивних карт.

Для створення інтерактивної карти використовується бібліотека Folium. Початковий крок – ініціалізація об'єкта карти з центром у середніх координатах усіх точок:

- `map_clusters = folium.Map(location=[data['latitude'].mean(), data['longitude'].mean()], zoom_start=6);`
- `location` задає координати центру карти (середнє значення широти і довготи всіх точок);
- `zoom_start` визначає рівень масштабування карти, що дозволяє охопити всі точки на початковому етапі.

Для покращення розуміння структури даних до карти можна додати легенду, яка пояснює, що означають кольори маркерів. Ось приклад реалізації:

```
def add_legend(map_clusters):
    legend_html = ""
    <div style="position: fixed;
        bottom: 50px; left: 50px; width: 200px; height: 150px;
        background-color: white; z-index:9999; font-size:14px; ">
    <b>Кластери:</b><br>
    <i style="background: #111111"></i> Кластер 1<br>
    <i style="background: #222222"></i> Кластер 2<br>
    ...
    </div>
    ""
    map_clusters.get_root().html.add_child(folium.Element(legend_html))
```

Ця функція додає блок HTML до карти, який пояснює, що означають різні кольори кластерів.

Для подальшого аналізу та звітності карту можна зберегти у форматі HTML:  
`map_clusters.save("clusters_map.html")`

Це дозволяє відкривати карту в будь-якому браузері та інтерактивно працювати з нею.

Відображення усіх доданих точок доставки на мапі відображено на рис.3.1.

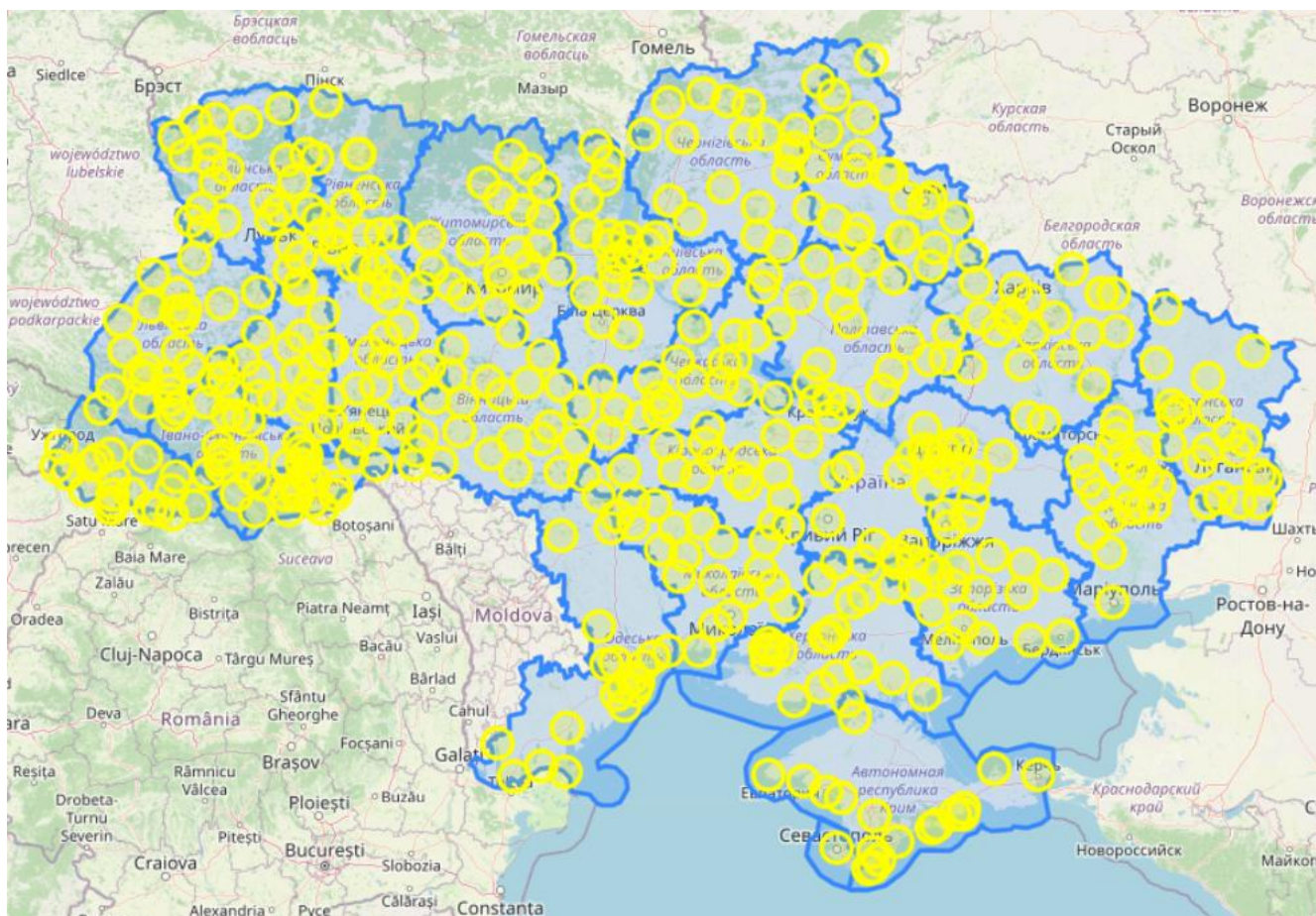


Рисунок 3.1 – Відображення усіх доданих точок доставки на мапі

За необхідності можна додати або видалити будь-яку точку на мапі використовуючи географічні координати у форматі XX.XXXX довгота, XX.XXXX широта, рис. 3.2.

Місто:

Широта:

Довгота:

Додати

Видалити

Рисунок 3.2 – Вікно взаємодії з базою даних

Якщо автоматично розраховані параметри вхідних даних для кластеризації не підходять під умови задачі, можна легко налаштувати їх за допомогою вікна налаштування параметрів, рис. 3.3.

### Налаштування параметрів вручну

Вхідні дані великогабарит K-Means	2	10	Застосувати
Вхідні дані малогабарит K-Means	2	10	Відновити
Вхідні дані великогабарит DBSCAN	0.2	2	Застосувати
Вхідні дані малогабарит DBSCAN	0.2	2	Відновити
Вхідні дані великогабарит Agglomerative	2		Застосувати
Вхідні дані малогабарит Agglomerative	2		Відновити
Вхідні дані великогабарит Spectral	2	nearest_neighbor	Застосувати
Вхідні дані малогабарит Spectral	2	nearest_neighbor	Відновити
Вхідні дані великогабарит Birch	2		Застосувати
Вхідні дані малогабарит Birch	2		Відновити

Рисунок 3.3 – Вікно налаштування параметрів вручну

Біля кожної кнопки додано впливаюче вікно з додатковою інформацією для допомоги користувачеві при використанні програми, рис. 3.4.

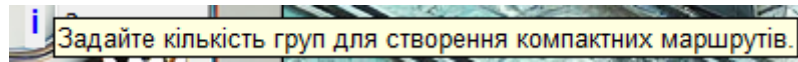


Рисунок 3.4 – Підказки для користувача

Результати проведених розрахунків оптимальних вхідних даних зберігаються, в даному випадку, в двох файлах – для кластеризації пунктів з рампою для вивантаження великогабаритного та важкого вантажу; для кластеризації пунктів без рампи, які можуть прийняти вантаж обмежений по вазі та габаритам, рис. 3.5. Дані параметри класифікації вантажу можна змінити при модифікації програми, застосовуючи будь-які додаткові характеристики для певних населених пунктів.

```

Файл  Редагувати  Переглянути
-----
Параметри: {'n_clusters': 7}, Метрики: {'silhouette': np.float64(0.3491936844946896),
'calinski_harabasz': np.float64(81.20757792602511), 'davies_bouldin':
np.float64(0.7804537062100441), 'density_variation': np.float64(0.17520633113102949)}
Параметри: {'n_clusters': 8}, Метрики: {'silhouette': np.float64(0.3502160723511202),
'calinski_harabasz': np.float64(85.98019734895277), 'davies_bouldin':
np.float64(0.8190934872313355), 'density_variation': np.float64(0.1312230661514665)}
Параметри: {'n_clusters': 9}, Метрики: {'silhouette': np.float64(0.34222038666813875),
'calinski_harabasz': np.float64(88.9205694523232), 'davies_bouldin':
np.float64(0.8738612202366522), 'density_variation': np.float64(0.12040424594116628)}
Параметри: {'n_clusters': 10}, Метрики: {'silhouette': np.float64(0.3532775054929227),
'calinski_harabasz': np.float64(89.13534353853842), 'davies_bouldin':
np.float64(0.8211765480474877), 'density_variation': np.float64(0.14939398174286245)}
Параметри: {'n_clusters': 11}, Метрики: {'silhouette': np.float64(0.35850237207560925),
'calinski_harabasz': np.float64(100.26636102361682), 'davies_bouldin':
np.float64(0.7884432321120766), 'density_variation': np.float64(0.1165727947458504)}
Параметри: {'n_clusters': 12}, Метрики: {'silhouette': np.float64(0.3864578485521106),
'calinski_harabasz': np.float64(111.69435193307427), 'davies_bouldin':
np.float64(0.7611206293796449), 'density_variation': np.float64(0.07741868785486179)}
Параметри: {'n_clusters': 13}, Метрики: {'silhouette': np.float64(0.3705663049930074),
'calinski_harabasz': np.float64(104.02617269064861), 'davies_bouldin':
np.float64(0.7319309644987235), 'density_variation': np.float64(0.35193381189146444)}
Параметри: {'n_clusters': 14}, Метрики: {'silhouette': np.float64(0.3705663049930074),
'calinski_harabasz': np.float64(104.0261726906486), 'davies_bouldin':
np.float64(0.7319309644986585), 'density_variation': np.float64(0.35193381189146444)}
Параметри: {'n_clusters': 15}, Метрики: {'silhouette': np.float64(0.3705663049930074),
'calinski_harabasz': np.float64(104.0261726906486), 'davies_bouldin':
np.float64(0.7319309644986585), 'density_variation': np.float64(0.35193381189146444)}

-----
Алгоритм | Найкращі параметри
-----
kmeans | {'n_clusters': 15, 'n_init': 20, 'random_state': 42}
dbscan | {'eps': np.float64(0.6000000000000001), 'min_samples': 4}
agglomerative | {'n_clusters': 13}
spectral | {'n_clusters': 2, 'random_state': 42, 'affinity': 'nearest_neighbors'}
birch | {'n_clusters': 12}
  
```

Рисунок 3.5 – Результат визначення найкращих параметрів та збереження їх у файли

Далі, для кластеризації і в подальшому побудові маршруту можна обрати будь-які точки для кластеризації. Результат, в даному випадку, відображається на мапі точок які мають: маркер-круг – ті, у яких наявна рампа (приймають великогабаритні та важкі вантажі); маркер-ромб – ті, які не мають рампи (приймають лише невеликі та легкі вантажі). Для наглядності візуалізації окремі кластери позначені окремими кольорами, рис. 3.6.

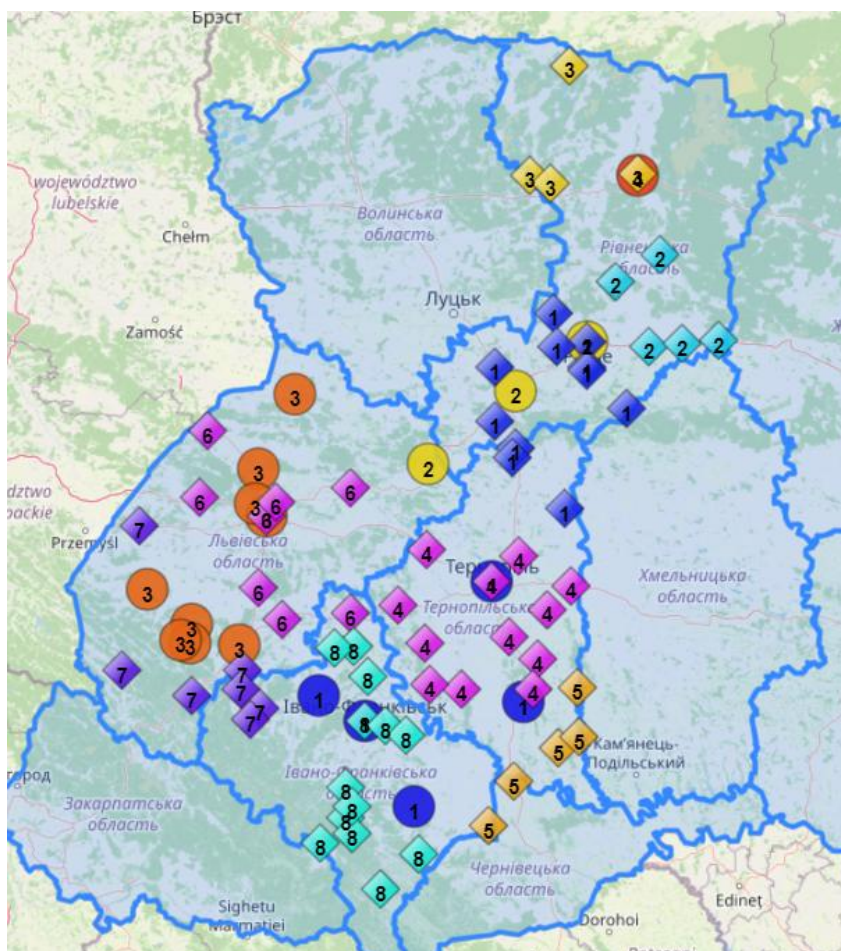


Рисунок 3.6 – Відображення результатів кластеризації на мапі

### 3.3 Засоби для побудови маршрутів на виробництві. Візуалізація маршрутів

Побудова маршрутів є ключовою частиною логістичної системи, оскільки дозволяє оптимізувати доставку вантажів, знижуючи витрати часу і ресурсів. У цьому підпункті розглянемо засоби, які використовуються для побудови

маршрутів, а також детально пояснимо роботу фрагмента коду, що відповідає за їх реалізацію.

Для побудови маршрутів у програмі використовується API Google Maps, що надає можливість враховувати реальні дорожні умови, включаючи відстань, тривалість маршруту та актуальну ситуацію на дорогах. Побудова маршрутів виконується для кожного кластера, що був сформований на етапі кластеризації.

У програмі реалізована функція `build_routes_for_clusters`, яка виконує побудову маршрутів для точок у кожному кластері. Розглянемо її роботу поетапно:

```
if clusters_with_ramp is not None and clusters_without_ramp is not None:
```

```
    def build_routes_for_clusters(cluster_dict, colors):
```

```
        routes = []
```

```
        for cluster_id, points in cluster_dict.items():
```

```
            if len(points) == 0:
```

```
                print(f"Cluster {cluster_id} has no points. Skipping.")
```

```
                continue
```

```
                print(f"Building routes for cluster {cluster_id} with {len(points)} points")
```

```
# Вивід інформації для налагодження
```

```
    color = colors[cluster_id % len(colors)]
```

```
    cluster_routes = build_route_with_google_maps(points, api_key, color)
```

```
    routes.append(cluster_routes)
```

```
    return routes
```

Логіка роботи функції:

- перевірка наявності кластерів:

```
if clusters_with_ramp is not None and clusters_without_ramp is not None,
```

перевіряється, чи існують кластери для точок із рампою та без рампи. Це гарантує, що побудова маршрутів буде виконана лише для доступних даних;

- ініціалізація списку маршрутів:

```
routes = [],
```

створюється порожній список, у який будуть додані маршрути для кожного кластера;

- ітерація по кластерах:

```
for cluster_id, points in cluster_dict.items(),
```

функція перебирає всі кластери, представлені у вигляді словника, де ключ – це ідентифікатор кластера, а значення – список точок у ньому;

- перевірка наявності точок у кластері:

```
if len(points) == 0:
```

```
    print(f"Cluster {cluster_id} has no points. Skipping.")
```

```
    continue,
```

якщо кластер не містить точок, його обробка пропускається, а інформація про це виводиться в консоль для налагодження;

- визначення кольору для маршруту:

```
color = colors[cluster_id % len(colors)],
```

кожному кластеру присвоюється колір із заданого списку `colors`. Це забезпечує наочність при візуалізації маршрутів;

- побудова маршрутів через API Google Maps:

```
cluster_routes = build_route_with_google_maps(points, api_key, color),
```

викликається функція `build_route_with_google_maps`, яка отримує список точок, API-ключ та колір. Ця функція повертає дані про маршрут, включаючи точки проходження, загальну відстань та час у дорозі;

- додавання маршрутів до списку:

```
routes.append(cluster_routes),
```

отриманий маршрут додається до загального списку `routes`.

- повернення результатів:

```
return routes,
```

після обробки всіх кластерів функція повертає список маршрутів, готових до візуалізації.

Отримані маршрути візуалізуються на інтерактивній карті за допомогою бібліотеки `Folium`. Для кожного маршруту використовується функція `folium.PolyLine`, яка відображає лінії між точками маршруту:

for route in routes:

```
folium.PolyLine(
    locations=route['path'],
    color=route['color'],
    weight=5,
    opacity=0.8
).add_to(map_clusters)
```

- locations задає координати точок маршруту.
- color визначає колір лінії маршруту.
- weight задає товщину лінії.
- opacity визначає прозорість.

Результат роботи функції побудови маршрутів показано на рис. 3.7.

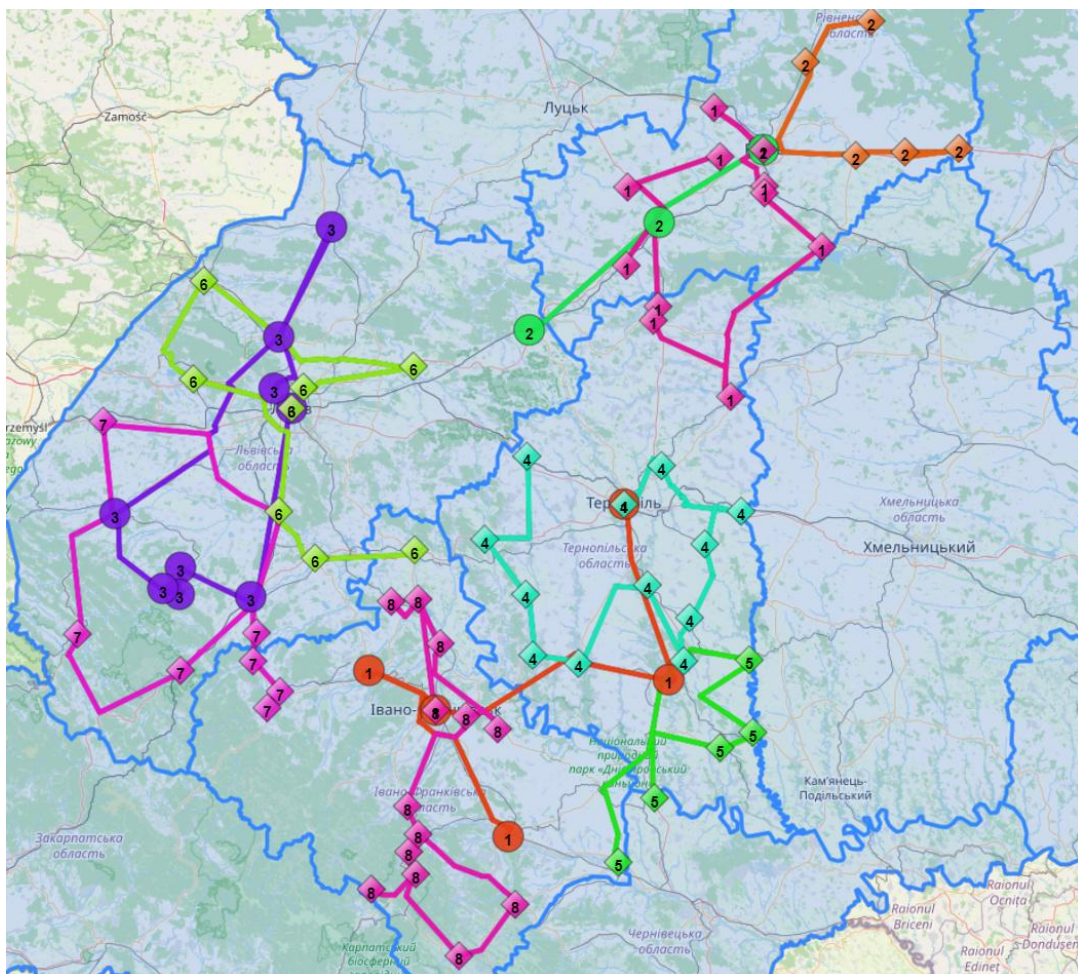


Рисунок 3.7 – Відображення побудованих маршрутів

Для побудови маршрутів у програмі використовується API Google Maps, що надає можливість враховувати реальні дорожні умови, включаючи відстань, тривалість маршруту та актуальну ситуацію на дорогах. Маршрути будуються з урахуванням географічних об'єктів, таких як гори, річки та інші природні перешкоди, що дозволяє уникати складних ділянок та оптимізувати логістику.

Візуалізація маршрутів на інтерактивній карті підвищує наочність даних і спрощує аналіз. Побудовані маршрути можуть бути використані для визначення оптимального розташування складів, розподільчих терміналів та інших важливих логістичних об'єктів.

### 3.4 Висновки до третього розділу

У третьому розділі було розглянуто застосування алгоритмів кластеризації, побудову маршрутів і їхню візуалізацію для підвищення ефективності логістичних процесів. У цьому розділі детально описано, як інструменти та технології, такі як API Google Maps і бібліотека Folium, дозволяють автоматизувати аналіз та оптимізацію маршрутів у реальних умовах.

Методи кластеризації забезпечили розподіл населених пунктів на групи залежно від їх географічного положення, характеристик вантажів і вимог до доставки. Це дало змогу знизити витрати та підвищити ефективність управління логістикою. Завдяки алгоритмам k-means, DBSCAN, Agglomerative, Spectral та Birch, вдалося знайти оптимальні параметри для кожного методу й адаптувати їх під конкретні задачі.

Побудова маршрутів за допомогою API Google Maps дозволила врахувати реальні дорожні умови, географічні перешкоди та актуальні обмеження. Це забезпечило точність маршрутів, зокрема у складних умовах, таких як міські райони або території з обмеженим доступом. Візуалізація результатів на інтерактивних картах надала зручний спосіб аналізу та презентації даних, що спрощує прийняття рішень.

Окремо варто зазначити, що побудовані маршрути можуть слугувати основою для визначення стратегічних точок розташування логістичних об'єктів, таких як склади або розподільчі термінали. Це сприяє подальшій оптимізації логістичних операцій та інтеграції системи у ширші бізнес-процеси.

Таким чином, проведені дослідження підтвердили, що використання сучасних алгоритмів кластеризації, побудови маршрутів і візуалізації є ефективними інструментами для підвищення продуктивності логістичних систем. Впровадження цих рішень забезпечує можливість адаптації до змінних умов та створює базу для подальшого розвитку інтегрованих логістичних систем.

## 4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ

### 4.1 Моделювання логістичних процесів у програмному середовищі

Моделювання розпочинається зі збору та структурування даних. Це включає інформацію про географічне розташування точок доставки, обсяг вантажів, часові обмеження та інші параметри, які мають вплив на логістичні процеси. Дані проходять попередню обробку, що включає перевірку на повноту, очищення та перетворення до формату, придатного для подальшого аналізу.

На основі підготовлених даних застосовуються алгоритми кластеризації для групування точок доставки за географічною близькістю, обсягом вантажу або іншими характеристиками. Результати кластеризації дозволяють розподілити ресурси таким чином, щоб мінімізувати витрати на транспортування та скоротити час доставки. Використання декількох методів кластеризації, таких як k-means, DBSCAN або Spectral, забезпечує адаптацію моделі до різних умов та типів даних.

Наступним етапом є побудова маршрутів для кожного сформованого кластера. У цьому процесі враховуються реальні дорожні умови, включаючи відстані між точками, наявність обмежень, а також географічні особливості регіону. API Google Maps забезпечує можливість побудови оптимальних маршрутів із урахуванням актуальної ситуації на дорогах, таких як затори чи закриття ділянок. Це дозволяє моделі враховувати динамічні зміни та коригувати маршрути в реальному часі.

Важливим елементом моделювання є візуалізація результатів. Інтерактивні карти, створені за допомогою спеціалізованих бібліотек, таких як Folium, відображають кластеризацію точок і побудовані маршрути. Завдяки цьому користувач отримує наочний інструмент для аналізу та оцінки роботи системи. Маршрути можуть бути відображені у вигляді ліній, що з'єднують точки доставки, із зазначенням ключових характеристик, таких як час і відстань.

Завершальним етапом є тестування моделі, під час якого оцінюється її точність та ефективність. На основі результатів моделювання формуються

рекомендації щодо покращення логістичних процесів. Наприклад, можуть бути визначені оптимальні місця для створення розподільчих центрів або запропоновані зміни в графіку доставки.

Перевагами моделювання у програмному середовищі є висока точність розрахунків, можливість адаптації до змінних умов та інтеграція з іншими системами управління. Крім того, використання цифрової моделі дозволяє зменшити витрати часу на експериментальне впровадження рішень і мінімізувати ризики, пов'язані з людським фактором.

Отже, моделювання логістичних процесів є невід'ємною частиною сучасного управління логістикою, яке сприяє підвищенню ефективності системи та забезпечує конкурентоспроможність підприємства на ринку.

#### 4.2 Перевірка працездатності системи на реальних даних підприємства

Перевірка працездатності системи на реальних даних підприємства дозволяє оцінити її ефективність у реальних умовах та визначити, наскільки вона відповідає практичним вимогам. Цей етап є важливим для підтвердження коректності роботи алгоритмів та здатності системи адаптуватися до змінних параметрів операцій.

Процес перевірки починається зі збору актуальних даних про логістичні процеси підприємства. Це включає координати точок доставки, обсяги та характеристики вантажів, часові обмеження для виконання доставки, а також інформацію про доступний транспортний парк, включаючи вантажопідйомність і витрати на експлуатацію транспортних засобів. Дані ретельно очищуються від можливих неточностей та дублювань для забезпечення максимальної точності моделювання.

На основі підготовлених даних система виконує розподіл точок доставки на кластери, враховуючи географічне положення та специфічні характеристики вантажів. Кластеризація дозволяє створити групи, які оптимально відповідають можливостям транспортних засобів, та значно спрощує побудову маршрутів. Для кожного кластера система автоматично генерує оптимальний маршрут,

враховуючи дорожні умови, наявність географічних обмежень і потребу мінімізувати час доставки.

Під час тестування аналізуються основні показники ефективності роботи системи. До них належать загальна довжина побудованих маршрутів, час виконання доставок, витрати на паливо та загальне завантаження транспортних засобів. На основі цих даних оцінюється, наскільки побудовані маршрути відповідають практичним цілям підприємства.

Результати тестування візуалізуються у вигляді інтерактивних карт, що демонструють маршрути та їхні ключові параметри. Це дозволяє легко ідентифікувати проблемні точки або зони для оптимізації. Візуалізація також допомагає визначити можливі місця для розташування додаткових складів або розподільчих центрів, які сприяли б подальшій оптимізації логістичних операцій.

Система має потенціал для розширення, включаючи можливість врахування додаткових параметрів вантажів, таких як температура зберігання, вимоги до вологості чи особливі умови транспортування. Такі властивості можуть бути інтегровані в модель, що підвищить її універсальність і придатність до специфічних логістичних завдань. Крім того, програмний продукт можна масштабувати: система ефективно працює як у межах одного міста, так і на рівні регіону або всієї країни.

Проведений аналіз підтвердив, що система здатна адаптуватися до специфічних умов підприємства, забезпечуючи ефективний розподіл ресурсів і зниження витрат на транспортування. Виявлені результати використовуються для подальшого вдосконалення логістичних процесів, включаючи перегляд графіка транспортування, підвищення ефективності завантаження складів і впровадження автоматизованого планування.

Таким чином, перевірка працездатності системи показала, що вона є надійним інструментом для управління логістичними операціями підприємства. Її впровадження сприяє підвищенню ефективності, скороченню витрат і підвищенню конкурентоспроможності на ринку.

#### 4.4 Рекомендації для застосування програми та можливі варіанти покращення

Розроблена програма є універсальним інструментом для оптимізації логістичних процесів. Її використання дозволяє значно підвищити ефективність управління перевезеннями, знизити витрати та забезпечити високу якість обслуговування клієнтів. Для ефективного впровадження програми важливо враховувати особливості конкретного підприємства, адаптуючи функціональність до його потреб.

Програму рекомендується інтегрувати з наявними системами управління підприємством, такими як ERP або CRM. Це забезпечить автоматичний імпорт даних про замовлення, маршрути та транспортні засоби, створюючи єдиний інформаційний простір для управління логістикою. Важливо також адаптувати програму до специфіки галузі. Наприклад, у харчовій промисловості необхідно враховувати температурний режим перевезення, а в фармацевтичній – умови зберігання продукції. Додавання спеціалізованих модулів дозволить підвищити універсальність програми.

Для забезпечення максимальної ефективності програмного забезпечення важливим етапом є навчання персоналу. Користувачі повинні розуміти базові принципи роботи програми, включаючи алгоритми кластеризації, побудови маршрутів і аналізу даних. Крім того, інтеграція з інструментами моніторингу в реальному часі дозволить оперативно реагувати на зміни в дорожніх умовах, графіках доставки або потребах клієнтів.

Програму можна суттєво покращити, інтегруючи сучасні технології. Використання IoT-пристроїв дозволить збирати дані в реальному часі про стан вантажів, місцезнаходження транспортних засобів і дорожню ситуацію, що сприятиме оперативному коригуванню маршрутів [27] - [29]. Додатково, впровадження алгоритмів штучного інтелекту допоможе аналізувати великі обсяги даних, прогнозувати затримки в доставці та оптимальні маршрути. Це особливо актуально для складних логістичних мереж.

Програму також можна розширити за рахунок аналітичних модулів, які дозволять оцінювати ефективність логістичних операцій у різних сценаріях. Наприклад, аналіз витрат на транспортування, прогнозування сезонних коливань у попиті або розрахунок оптимального завантаження транспортних засобів допоможуть приймати обґрунтовані управлінські рішення. Масштабування програми до рівня міжнародної логістики дозволить враховувати митні правила, вартість транспортування та доступність інфраструктури в різних країнах.

Для підвищення зручності взаємодії з клієнтами можна інтегрувати мобільні додатки, що дозволять відстежувати статус замовлень у реальному часі, отримувати сповіщення та взаємодіяти з операторами. Крім того, автоматизація фінансового обліку допоможе спростити розрахунок витрат на транспортування, а також генерувати прозорі звіти для бухгалтерії.

Одним із важливих напрямів розвитку є врахування екологічного аспекту. Програма може пропонувати маршрути, які мінімізують вплив на довкілля, наприклад, оптимізуючи споживання палива або використовуючи електромобілі. Це не лише знижує витрати, але й сприяє підвищенню іміджу підприємства.

Таким чином, програма має великий потенціал для впровадження та розвитку в різних галузях і масштабах. Вона здатна не лише оптимізувати логістичні процеси, але й інтегрувати сучасні технології для забезпечення адаптивності та універсальності. Масштабованість програми дозволяє застосовувати її як у межах міста, так і на рівні міжнародної логістики, що робить її цінним інструментом для розвитку бізнесу.

#### 4.5 Висновки до четвертого розділу

У четвертому розділі було розглянуто ключові аспекти моделювання логістичних процесів, перевірки працездатності системи на реальних даних та надано рекомендації для її вдосконалення. Розроблена система довела свою ефективність як інструмент для автоматизації управління логістичними операціями, дозволяючи адаптувати її до специфічних потреб підприємств.

Моделювання логістичних процесів у програмному середовищі дало змогу створити інтегровану модель, яка забезпечує аналіз, оптимізацію та управління маршрутами з урахуванням реальних дорожніх умов і ресурсних обмежень. Завдяки використанню алгоритмів кластеризації та побудови маршрутів, система спрощує розподіл вантажів між транспортними засобами та забезпечує їх ефективне використання.

Під час перевірки працездатності системи на реальних даних підприємства підтверджено її здатність адаптуватися до змінних умов і забезпечувати точні розрахунки. Тестування виявило можливості для зниження витрат на транспортування, скорочення часу доставки та підвищення якості обслуговування клієнтів. Аналіз ефективності системи виявив, що результати моделювання можуть бути використані для стратегічного планування, включаючи вибір місця розташування складів і оптимізацію графіків доставки.

Рекомендації щодо впровадження програми включають інтеграцію з існуючими системами управління підприємством, адаптацію до галузевих вимог і навчання персоналу. Програма має високий потенціал для вдосконалення, зокрема за рахунок впровадження IoT-пристроїв, використання штучного інтелекту та аналітичних модулів для прогнозування. Крім того, можливість масштабування системи робить її придатною для використання як у межах міста, так і в міжнародній логістиці.

Таким чином, четвертий розділ демонструє, що розроблена система є не лише ефективним інструментом для оптимізації логістичних процесів, але й базою для подальшого розвитку та вдосконалення. Її використання сприяє автоматизації рутинних операцій, підвищенню точності прийняття рішень і створенню конкурентних переваг для підприємств у сучасних умовах ринку.

## 5 ОХОРОНА ПРАЦІ

### 5.1 Аналіз умов праці на транспортних підприємствах

Аналіз умов праці на транспортних підприємствах є важливою складовою забезпечення охорони праці. Ця галузь характеризується підвищеними ризиками для здоров'я та безпеки працівників через складні умови роботи, високу інтенсивність операцій і постійний вплив зовнішніх факторів.

На транспортних підприємствах працівники стикаються з фізичними, хімічними, біологічними та психофізіологічними факторами ризику. Основними фізичними факторами є шум, вібрація, перепади температур і робота в умовах недостатнього освітлення. Ці фактори можуть негативно впливати на здоров'я працівників, викликаючи підвищену стомлюваність, зниження концентрації уваги та розвиток професійних захворювань.

Хімічні фактори ризику включають вплив вихлопних газів, паливно-мастильних матеріалів і хімічних речовин, які використовуються для обслуговування транспорту. Довготривалий контакт із цими речовинами може призвести до захворювань дихальної системи, шкіри та інших органів.

Біологічні фактори ризику, такі як інфекції або контакти з алергенами, найчастіше виникають у працівників, які виконують завдання з перевезення харчових продуктів або медичних вантажів. Ці умови вимагають додаткових заходів для захисту працівників, включаючи використання індивідуальних засобів захисту та регулярний медичний огляд.

Психофізіологічні фактори включають високу емоційну напруженість, пов'язану з відповідальністю за безпеку вантажів і пасажирів, необхідністю дотримання жорстких графіків і роботу в умовах постійного стресу. Усе це може призводити до розвитку синдрому професійного вигорання, зниження продуктивності праці та підвищення ймовірності нещасних випадків.

Статистика травматизму на транспортних підприємствах свідчить про необхідність постійного моніторингу умов праці та впровадження превентивних

заходів. Більшість нещасних випадків пов'язані з порушенням техніки безпеки, недоліками в організації робочого процесу або використанням несправного обладнання. Професійні захворювання найчастіше виникають унаслідок тривалого впливу шкідливих факторів без належного захисту.

Для покращення умов праці на транспортних підприємствах необхідно впроваджувати сучасні технології, які знижують вплив шкідливих факторів. Наприклад, використання транспорту з низьким рівнем шуму та вібрації, автоматизація рутинних процесів, впровадження екологічно чистих паливних систем. Одним із ключових моментів є впровадження сучасних інформаційних систем для моніторингу дотримання норм охорони праці, що дозволяє знижувати ризики шляхом оперативного реагування на можливі небезпеки [30].

Додатково слід забезпечити працівників якісними засобами індивідуального захисту, такими як шумопоглинаючі навушники, спеціальний одяг і рукавички. Важливо також організувати регулярне навчання працівників із техніки безпеки та надання першої медичної допомоги.

Поліпшення умов праці також включає психологічну підтримку працівників. Введення програм управління стресом, надання консультацій із психологами та організація відпочинку для працівників, які працюють у напружених умовах, сприятиме підвищенню продуктивності та зниженню рівня професійного вигорання.

## 5.2 Рекомендації щодо поліпшення умов праці

Поліпшення умов праці на транспортних підприємствах є важливим завданням, яке сприяє збереженню здоров'я працівників, підвищенню продуктивності та мінімізації ризиків, пов'язаних із професійними захворюваннями і травматизмом. Для досягнення цих цілей необхідно впроваджувати комплексний підхід, який охоплює технічні, організаційні, соціальні та екологічні аспекти.

Технічні заходи включають модернізацію робочих місць за допомогою сучасного обладнання та транспорту, які мають низький рівень шуму, вібрації та працюють на екологічно чистому паливі. Регулярне технічне обслуговування транспорту дозволяє уникати несправностей і знижує ризик аварій. Використання інноваційних матеріалів, таких як антивібраційні покриття або шумозахисні панелі, значно покращує комфорт працівників.

Організаційні заходи спрямовані на підвищення ефективності управління та дотримання правил безпеки. Серед них важливе місце займають регулярні тренінги для працівників, оптимізація робочих графіків із забезпеченням належного часу для відпочинку, а також внутрішні аудити умов праці. Такі заходи дозволяють зменшити ризики, пов'язані з порушенням техніки безпеки, і забезпечити своєчасне реагування на потенційні проблеми.

Соціальний аспект включає заходи, які сприяють збереженню психологічного здоров'я працівників. Надання доступу до психологічних консультацій, створення програм мотивації та облаштування комфортних зон відпочинку допомагають зменшити рівень стресу і професійного вигорання. Це позитивно впливає на загальну атмосферу в колективі та продуктивність роботи.

Екологічні рішення є важливою складовою забезпечення якісних умов праці. Оптимізація маршрутів транспорту знижує витрати на паливо та мінімізує вплив на довкілля. Впровадження електротранспорту та будівництво зарядних станцій сприяє переходу до більш екологічної логістики. Також важливим є запровадження систем утилізації відходів, які генеруються підприємством, для зменшення негативного впливу на природу.

Для впровадження цих рекомендацій необхідно розробити детальний план дій із зазначенням відповідальних осіб і строків виконання. Моніторинг ефективності реалізованих заходів допоможе вчасно вносити корективи та забезпечити досягнення запланованих результатів. Важливим аспектом є також фінансова підтримка, яка може бути забезпечена через бюджет підприємства або зовнішні джерела, такі як грантові програми.

Загалом, виконання технічних, організаційних, соціальних і екологічних рекомендацій створить безпечне та комфортне робоче середовище для працівників транспортних підприємств. Це сприятиме не лише зменшенню рівня травматизму, але й підвищенню продуктивності праці та якості роботи підприємства загалом.

## ВИСНОВКИ

У результаті проведених досліджень розроблено інтегровану логістичну систему, яка забезпечує автоматизацію процесів кластеризації, побудови маршрутів і візуалізації даних. Система довела свою ефективність у моделюванні та оптимізації логістичних операцій з урахуванням реальних умов.

Позитивним результатом роботи стало використання сучасних алгоритмів кластеризації, таких як k-means, DBSCAN, Agglomerative, Spectral і Birch, що дозволило забезпечити адаптивність системи до різних умов. Це сприяло зниженню витрат на транспортування, скороченню часу доставки та підвищенню точності планування маршрутів.

Перевірка працездатності системи на реальних даних підприємства підтвердила її здатність адаптуватися до специфічних умов та забезпечувати ефективний розподіл ресурсів. Водночас було виявлено, що обмеження системи полягають у необхідності ручного налаштування деяких параметрів, що може бути покращено шляхом автоматизації цього процесу.

Основним недоліком розробленої системи є недостатня інтеграція додаткових параметрів вантажів, таких як температурні режими чи особливі вимоги до транспортування. Для подальшого вдосконалення системи рекомендується розширити функціонал, включивши можливість врахування унікальних характеристик вантажів.

Запропоновані технології візуалізації на основі інтерактивних карт значно полегшують аналіз отриманих даних, проте потребують більшого розвитку у частині створення автоматизованих звітів. Додавання модулів для генерації звітів дозволить підвищити зручність роботи користувачів із системою.

Для підвищення ефективності системи рекомендовано впровадження IoT-технологій та використання штучного інтелекту для аналізу великих обсягів даних у реальному часі. Це дозволить підвищити точність прогнозування, оперативність прийняття рішень та автоматизацію управління логістичними процесами.

Результати роботи можуть бути масштабовані для використання як на рівні окремого міста, так і на національному рівні. Це підтверджує універсальність розробленої системи та її здатність адаптуватися до різних умов і задач логістики.

У процесі дослідження підтверджено важливість інтеграції екологічних рішень у логістичні системи, зокрема шляхом оптимізації маршрутів для зниження викидів шкідливих речовин і впровадження електротранспорту. Рекомендовано продовжити роботу над екологічною складовою для підвищення конкурентоспроможності системи.

Загалом виконана робота демонструє високий потенціал запропонованих рішень для оптимізації логістичних процесів. Подальший розвиток системи може бути спрямований на інтеграцію з іншими інформаційними платформами, розширення функціоналу для врахування специфіки різних галузей та поліпшення користувацького досвіду.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Методичні вказівки з підготовки та захисту кваліфікаційної роботи здобувачами другого (магістерського) рівня вищої освіти спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології, освітньо-професійних програм: «Автоматизоване управління технологічними процесами», «Комп'ютерно-інтегровані технологічні процеси і виробництва», «Комп'ютеризовані та робототехнічні системи» / Упоряд. І. Ш. Невлюдов, Р. В. Артюх, В. В. Безкоровайний, Н. П. Демська, В. В. Євсєєв, О. І. Филипенко, О. М. Цимбал. – Харків: ХНУРЕ, 2021. – 55 с.

2. ДСТУ 3008-15. Інформація та документація. Звіти у сфері науки та техніки. Структура та правила оформлювання. / Нац. стандарт України. – Вид. офіц. – [Чинний від 2015-06-22]. – Київ: Держстандарт України, 2017. – 26 с.

3. Каленюк І. П. Управління логістичними процесами: підручник / за ред. І. П. Каленюк. – Київ: Наукова думка, 2020. – 350 с.

4. Smith J. Logistics Clustering Strategies: A Data-Driven Approach / Smith J., Lee K. // International Journal of Logistics Management. – 2022. – Vol. 33, No. 2. – P. 178–193.

5. Brown T. Spectral Clustering Applications in Logistics / Brown T., Zhang L. // Computational Logistics Journal. – 2021. – Vol. 12, No. 3. – P. 345–362.

6. Zhao L., Wang R. BIRCH Algorithm for Dynamic Data Clustering / Zhao L., Wang R. // Data Mining Techniques Journal. – 2020. – Vol. 8, No. 1. – P. 50–67.

7. Johnson S. Logistic Clustering Strategies for Optimization / Johnson S. // International Journal of Logistics Research. – 2021. – Vol. 28, No. 3. – P. 123–140.

8. Smith J. Advanced Routing and Clustering Algorithms in Logistics / Smith J. // Computational Logistics Journal. – 2022. – Vol. 35, No. 4. – P. 450–472.

9. Zhao L. Clustering Techniques in Dynamic Logistics Systems / Zhao L. // Data Science and Logistics. – 2020. – Vol. 18, No. 2. – P. 78–95.

10. Невлюдов І.Ш. "Теорія автоматичного управління (збірник задач)": навчальний посібник для студентів спеціальності 151 "Автоматизація та комп'ютерно-інтегровані технології"/ І.Ш. Невлюдов, О.В. Токарєва; Харків: ХНУРЕ, 2020. - 240 с. - ISBN 978-617-7859-57-3
11. Smith J. Advanced Programming Techniques in Python for Logistics Applications / Smith J. // Computational Logistics Journal. – 2022. – Vol. 35, No. 3. – P. 450–472.
12. Hahn R. Data Analytics and Python Integration in Logistics Systems / Hahn R. // Modern Data Science Journal. – 2020. – Vol. 17, No. 4. – P. 89–105.
13. Johnson S. Python Applications in Real-Time Logistics Modeling / Johnson S. // International Journal of Logistics Research. – 2021. – Vol. 28, No. 5. – P. 123–140.
14. Carter L. MySQL for Data Management in Logistics / Carter L. // Database Trends and Applications. – 2021. – Vol. 19, No. 2. – P. 45–60.
15. Євсєєв В.В. Проектування мобільних роботів на базі одноплатних комп'ютерів (Raspberry Pi и мови Python 3.6) // Невлюдов І. Ш., Андрусевич А. О., Євсєєв В. В. Підручник. – Харків : 2020. С. 257.
16. Кластеризація в управлінні глобальними ланцюгами постачання [Електронний ресурс] // DSI Management. – 2023. – Режим доступу: <https://dsim.khmnu.edu.ua/index.php/dsim/article/view/73>. – 12.12.2024.
17. Гаврилюк О. О. Методи кластеризації в задачах логістики / О. О. Гаврилюк // Вісник НТУУ "КПІ". - 2021. - №3. - С. 45–52.
18. Петренко С. П. Застосування алгоритмів кластеризації для логістичних систем / С. П. Петренко // Український журнал інформаційних технологій. - 2022. - №2. - С. 34–41.
19. Smith J. Advanced Machine Learning Applications in Logistics. / Smith J., Johnson L. - Academic Press, 2020. - 560 p.
20. Brown P. Clustering Techniques for Transportation Logistics / Brown P. // Logistics Journal. - 2018. - Vol. 15. No. 4. - P. 321–330.
21. Jones R. Neural Networks for Demand Prediction in Supply Chains / Jones R., Taylor K., Green D. - Elsevier, 2019. - 432 p.

22. White S. Multi-Criteria Optimization in Logistics / White S., Harris M. - Springer, 2022. - 380 p.
23. Christopher M. Logistics & Supply Chain Management / Christopher M. - Financial Times Press, 2016. - 488 p.
24. Rushton A. The Handbook of Logistics and Distribution Management / Rushton A., Croucher P., Baker P. - Kogan Page, 2017. - 720 p.
25. Fleischmann B. Advanced Planning in Supply Chain Management / Fleischmann B., Meyr H., Wagner M. - Springer, 2003. - 370 p.
26. Теорія автоматичного управління (збірник задач) [Текст]: навч. посіб. для студентів спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології / І.Ш. Невлюдов, О.В. Токарева; Харків. нац. ун-т радіоелектроніки. - Харків: Панов А.М., 2020. – 240 с.
27. Nevliudov, I., & et al.. (2020). Method of Algorithms for Cyber-Physical Production Systems Functioning Synthesis. International Journal of Emerging Trends in Engineering Research, 8(10), 7465-7473
28. Невлюдов І.Ш. Виробничі процеси та обладнання об'єктів автоматизації. Збірник задач: Навчальний посібник / І.Ш. Невлюдов, А.О. Андрусевич, Г.В. Пономарьова, А.О. Функендорф. Кривий Ріг: КК НАУ. 2018. – 332 с.
29. Невлюдов І.Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 1: підручник. Харків: ФОП Панов А.М., 2021. – 604 с.
30. Стасюк А. І. Охорона праці в галузі: підручник / А. І. Стасюк, В. П. Мельник. – Київ: Освіта України, 2019. – 320 с.