

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу**

«На правах рукопису»
УДК 519.854

«До захисту допущено»

Завідувач кафедри

_____ О.Л. Тимошук

«__» _____ 20__ р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо – науковою програмою

зі спеціальності 124 Системний аналіз

**на тему: «Метод підвищення ефективності розміщення мобільних
транспортних засобів у населеному пункті»**

Виконала:

студентка II курсу, групи КА – 92мп

Олесюк Інна Юріївна _____

Керівник:

професор кафедри ММСА, д.т.н., проф.

Мухін В.Є. _____

Рецензент:

доц. кафедри технічної кібернетики

КПІ ім.Ігоря Сікорського,

д.т.н., доц.

Корнага Я.І. _____

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.

Студентка _____

Київ
2020

**Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»
Інститут прикладного системного аналізу
Кафедра математичних методів системного аналізу**

Рівень вищої освіти – другий (магістерський)

Спеціальність – 124 «Системний аналіз»

ЗАТВЕРДЖУЮ
Завідувач кафедри

(підпис)

(ініціали, прізвище)

«__» _____ 20__ р.

**ЗАВДАННЯ
на магістерську дисертацію студентці
Олесюк Інни Юріївни**

1. Тема дисертації: «Метод підвищення ефективності розміщення мобільних транспортних засобів у населеному пункті», науковий керівник дисертації Мухін Вадим Євгенович, д.т.н., професор, затверджені наказом по університету від «02» листопада 2020 р. № 3182-с.

2. Термін подання студентом: 12.12.20.

3. Об'єкт дослідження: Маршрутизація в умовах обмеженої вантажомісткості.

4. Предмет дослідження: Методи ребалансування транспорту в системах спільного використання мобільних транспортних засобів.

5. Перелік завдань, які потрібно розробити:

1. Здійснити огляд технічної літератури за темою роботи;
2. Дослідити актуальність обраної теми;
3. Ознайомитись із існуючими методами ребалансування для систем спільного використання мобільних транспортних засобів;

4. Здійснити порівняльний аналіз наявних методів, проаналізувати їх переваги та недоліки;
5. Запропонувати і реалізувати метод ребалансування систем мобільних транспортних засобів;
6. Провести аналіз результатів;
7. Провести аналіз ринкових можливостей запуску стартап проекту;
8. Розробити концептуальні висновки;
9. Підготувати ілюстративний матеріал;
10. Оформити пояснювальну записку.

6. Орієнтовний перелік ілюстративного матеріалу:

1. Постановка завдання дослідження;
2. Існуючі методи ребалансування для систем мобільних транспортних засобів, аналіз переваг та недоліків;
3. Практичний приклад роботи прикладної програми, розробленої в ході дослідження;
4. Структура програмного продукту;
5. Розрахункові таблиці з результатами.

7. Орієнтовний перелік публікацій:

1. “A method of enhancing the mobile vehicles deployment effectiveness in built – up areas” - scientific journal “Science of Europe”. Vol 2, № 59. P. 46-52.
2. “Підхід до прогнозування попиту для задачі статичного ребалансування” - матеріали XXXII Міжнародної науково-практичної інтернет-конференції «Проблеми та перспективи розвитку сучасної науки в країнах Європи та Азії» (Переяслав, 30 листопада 2020 р.). С. 67-70.

8. Дата видачі завдання: 5 вересня 2019 року.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Отримання завдання на магістерську дисертацію	01.09.2020 – 06.09.2020	
2.	Огляд технічної літератури за темою	07.09.2020 – 13.09.2020	
3.	Концептуальний вступ дисертації. Формулювання об'єкта, предмета, цілі, завдань, новизни, практичної значущості результатів. Дослідження актуальності обраної теми	14.09.2020 – 20.09.2020	
4.	Перший розділ. Аналіз існуючих методів для ребалансування	21.09.2020 – 27.09.2020	
5.	Другий розділ. Аналіз підходів прогнозування попиту. Побудова математичної моделі для знаходження оптимального маршруту перепозиціонування	28.09.2020 – 04.10.2020	
6.	Розробка початкового програмного забезпечення	05.10.2020 – 11.10.2020	
7.	Третій розділ. Покращення якості програмного забезпечення	12.10.2020 – 18.10.2020	
8.	Аналіз результатів	19.10.2020 – 25.10.2020	
9.	Проведення аналізу ринкових можливостей стартап – проекту	26.10.2020 – 01.11.2020	
10.	Підготовка ілюстративного матеріалу	02.11.2020 – 15.11.2020	
11.	Оформлення пояснювальної записки	16.11.2020 – 27.11.2020	

Студент _____

І.Ю. Олесюк

Науковий керівник дисертації _____

В.Є.Мухін

РЕФЕРАТ

Магістерська дисертація: 119 с., 34 рис., 34 табл., 1 додаток, 23 джерела.

Об'єктом дослідження є маршрутизація в умовах обмеженої вантажомісткості.

Предмет дослідження – методи ребалансування транспорту в системах спільного використання мобільних транспортних засобів.

Методами досліджень виступають методи статичного та динамічного ребалансування.

Мета даної роботи полягає у тому щоб дослідити та вдосконалити існуючі методи ребалансування та розробити систему, що здійснюватиме дану процедуру.

Актуальність теми: у всьому світі спостерігається активне впровадження систем спільного використання мобільних транспортних засобів. Такі системи забезпечують недорогу та екологічну альтернативу транспорту, вирішуючи багато проблем міста. Однак, їх швидке розповсюдження призвело до безлічі проблем: пошкодження велосипедів, потреба у паркуванні та обслуговуванні транспорту, постійна наявність велосипедів на станціях. Ключем до вирішення усіх цих проблем є вдале ребалансування.

Результати роботи: запропоновано новий метод ребалансування, що складається із етапу прогнозування та перепозиціонування. Практичним результатом є розробка системи ребалансування мобільних транспортних засобів.

Новизна роботи: запропоновано новий метод для проведення статичного ребалансування, котрий відрізняється від наявних тим, що при проведенні операції перепозиціонування враховує наявність пошкоджених транспортних засобів у системі, а також ціну використання конкретного автомобіля.

ЗАДАЧА МАРШРУТИЗАЦІЇ ТРАНСПОРТУ, СТАТИЧНЕ РЕБАЛАНСУВАННЯ, ТРАНСПОРТНА ЛОГІСТИКА, ПРОГНОЗУВАННЯ, ГРАДІЄНТ БУСТИНГ, ЦІЛОЧИСЕЛЬНЕ ПРОГРАМУВАННЯ

ABSTRACT

Master's thesis explanatory note: 119 p., 34 tables, 34 fig., 1 application, 23 references.

The object of the research is a routing problem with restricted loading capacities.

Subject of the research – methods of rebalancing transport in mobile vehicles sharing systems.

Research methods are static and dynamic rebalancing methods.

The purpose of this work is to research and improve existing methods of rebalancing for mobile vehicles sharing systems and to develop a system that will carry out this procedure.

Relevance of the topic: Nowadays bike-sharing systems are extremely popular all over the world. They have been installed in many cities. They provide a low-cost and environment-friendly transportation alternative and help reduce traffic congestion. However, the rapid spread of such systems has led to many problems. Main problems are large number of damaged bicycles in the system, parking and transport services problems, all – time availability of bicycles at the stations. The key to solving all problems mentioned above is a successful rebalancing.

Results of the research: a new method of rebalancing, which consists of two stages – forecasting and repositioning is proposed. The practical result is the development of a system for rebalancing mobile vehicles.

The novelty of the work: a new method of rebalancing is proposed. This method considers a number of damaged bicycles and price of usage definite vehicle for rebalancing.

VEHICLE ROUTING PROBLEM, STATIC REBALANCING, TRANSPORT LOGISTICS, PREDICTION, GRADIENT BOOSTING, MIXED – INTEGER LINEAR PROGRAMMING

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ _____	10
ВСТУП _____	11
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ЗАДАЧІ МАШРУТИЗАЦІЇ ТРАНСПОРТУ _____	13
1.1 Опис задачі маршрутизації транспорту _____	13
1.2 Задача маршрутизації транспортного засобу із підбором та доставкою _____	14
1.3 Аналіз актуальності задачі розміщення мобільних транспортних засобів _____	16
1.4 Опис проблеми ребалансування мобільних транспортних засобів _____	19
1.5 Аналіз існуючих підходів до вирішення проблеми ребалансування мобільних транспортних засобів _____	21
1.5.1 Стратегічне планування _____	21
1.5.2 Статичне ребалансування _____	25
1.5.3 Динамічне ребалансування _____	28
1.5.4 Прогнозування попиту на станціях для ребалансування _____	32
1.6 Висновки _____	35
РОЗДІЛ 2 МЕТОД СТАТИЧНОГО РЕБАЛАНСУВАННЯ ДЛЯ СИСТЕМ СПІЛЬНОГО ВИКОРИСТАННЯ МОБІЛЬНИХ ТРАНСПОРТНИХ ЗАСОБІВ _____	36
2.1 Задача статичного ребалансування мобільних транспортних засобів _____	36
2.2 Прогнозування попиту _____	38
2.2.1 Визначення патернів поведінки станцій _____	38
2.2.2 Визначення попиту _____	40
2.2.3 Прогнозування максимального щоденного попиту станції _____	42
2.2.4 Виключення збалансованих станцій _____	46

	8
2.3	Математична модель задачі перерозподілу транспортних засобів_ 47
2.3.1	Початкова математична модель _____ 47
2.3.2	Врахування додаткових умов та обмежень _____ 50
2.4	Загальний опис методу статичного ребалансування_____ 54
2.5	Критерій якості ребалансування _____ 55
2.6	Висновки _____ 56
РОЗДІЛ 3 РОЗРОБКА СТРУКТУРИ ПРОГРАМНОГО КОМПЛЕКСУ _____ 58	
3.1	Обрані технології _____ 58
3.2	Структура програмного продукту_____ 59
3.2.1	Структура системи прогнозування _____ 60
3.2.2	Структура системи перепозиціонування _____ 61
3.3	Знаходження рішення задачі із врахуванням кількості поломаних транспортних засобів _____ 62
3.3.1	Загальний огляд методу _____ 62
3.3.2	Представлення почтакового рішення _____ 62
3.3.3	Зміна швидкості та позиції частинки _____ 64
3.3.4	Застосування пошуку зі змінними околами _____ 64
3.4	Знаходження рішення задачі з врахуванням характеристик автомобілів для ребалансування _____ 66
3.5	Висновки _____ 66
РОЗДІЛ 4 АНАЛІЗ ПРАКТИЧНОГО ДОСЛІДЖЕННЯ_____ 68	
4.1	Дані _____ 68
4.2	Прогнозування попиту на станціях _____ 70
4.3	Здійснення перепозиціонування _____ 75
4.4	Висновки _____ 82
РОЗДІЛ 5 РОЗРОБЛЕННЯ СТАРТАП – ПРОЕКТУ _____ 84	
5.1	Опис ідеї стартап проекту _____ 84
5.2	Технологічний аудит ідеї проекту _____ 85
5.3	Аналіз ринкових можливостей запуску стартап проекту _____ 86
5.3.1	Аналіз попиту та потенційних груп клієнтів _____ 86

	9
5.3.2	Аналіз ринкового середовища _____ 88
5.3.3	Аналіз пропозиції _____ 90
5.3.4	Фактори конкурентоспроможності _____ 93
5.4	Розроблення ринкової стратегії проекту _____ 97
5.4.1	Вибір цільових груп потенційних споживачів _____ 97
5.4.2	Базова стратегія розвитку _____ 98
5.4.3	Стратегія конкурентної поведінки _____ 99
5.4.4	Стратегія позиціонування _____ 100
5.5	Розроблення маркетингової програми стартап – проекту _____ 101
5.5.1	Ключові переваги концепції потенціоного товару _____ 101
5.5.2	Трирівнева маркетингова модель товару _____ 102
5.5.3	Визначення цінових меж _____ 103
5.5.4	Формування системи збуту _____ 103
5.6	Висновки _____ 106
ВИСНОВКИ _____ 107	
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ _____ 109	
ДОДАТОК А ЛІСТИНГ ОСНОВНОГО МОДУЛЮ ПРОГРАМНОГО ПРОДУКТУ _____ 112	

ПЕРЕЛІК СКОРОЧЕНЬ

ССВМТЗ – система спільного використання мобільних транспортних засобів;

ЗМТ – задача маршрутизації транспортних засобів;

ТЗ – транспортні засоби;

МТЗ – мобільні транспортні засоби.

ВСТУП

Проблема маршрутизації транспорту є одним із класів задач логістики. Метою вирішення таких задач є побудова маршрутів для одного або декількох ТЗ, які будуть вважатися оптимальними стосовно заданих критеріїв. В даний час питання оптимізації транспортної логістики є дуже важливим для більшості підприємств. Від вирішення цього завдання безпосередньо залежить ціна на товар, більш того в деяких областях ринку витрати на доставку товару пропорційні його вартості.

У всьому світі останні декілька років спостерігається активне впровадження ССВМТЗ. Ці системи пропонують послуги, в яких громадськості надається транспорт для швидкого пересування доступний для спільного користування. ССВМТЗ є надзвичайно корисними для інфраструктури кожного міста та є важливим інструментом, що використовується державними адміністраціями для отримання більш стійкої мобільності, зменшення інтенсивності дорожнього руху, усунення завантаження та екологічного й шумового забруднення спричинених транспортними перевезеннями. В силу різноманітності подорожей клієнтів такі системи потребують постійного ребалансування вкінці робочого дня, що є досить дорогою операцією з точки зору логістики. Саме неефективність цієї операції є однією з основних причин невдоволення клієнтів, а також основною перешкодою до встановлення таких систем приватними та державними компаніями. Тому, наразі, досить важливим є наявність автоматизованого рішення даної проблеми. Актуальність дослідження зумовлена зростаючою кількістю таких систем у всьому світі.

Метою даної роботи є дослідження та вдосконалення існуючих методів ребалансування, а також розробка системи, що здійснюватиме дану процедуру.

В ході виконання магістерської дисертації було розроблено новий метод для проведення статичного ребалансування, котрий відрізняється від наявних тим, що при проведенні операції перепозиціонування враховує наявність

пошкоджених транспортних засобів у системі, а також ціну використання конкретного автомобіля.

У першому розділі розкриваються загальні теоретичні відомості щодо ЗМТ. Детально розглядається задача ребалансування та її актуальність. Особлива увага приділяється існуючим практикам, методам та доцільності їх застосування. Розглядаються різні можливі підходи до виконання перепозиціонування та прогнозування попиту на станціях ССВМТЗ.

Другий розділ присвячено розробленому методу статичного ребалансування для ССВМТЗ. Саме в цьому розділі описуються математичні основи для прогнозування та перепозиціонування транспорту на станціях. А саме, було наведено детальний опис підходу до прогнозування попиту на станціях та представлена математична модель для знаходження оптимального маршруту репозиціонування. Також наявний детальний опис розробленого методу статичного ребалансування.

У третьому розділі описуються деталі розробки та представлена структура програмного продукту.

Четвертий розділ присвячено аналізу практичного дослідження реалізованого методу, а саме проведено аналіз результатів прогнозування та перепозиціонування. Наведене порівняння реалізованих підходів з іншими.

Останній розділ роботи стосується опису стартап проекту, його аналізу. Основні думки щодо проведеного аналізу та пропозиції відносно подальшого розвитку проєкту описуються у висновках.

РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ЗАДАЧІ МАШРУТИЗАЦІЇ ТРАНСПОРТУ

1.1 Опис задачі маршрутизації транспорту

Задача маршрутизації транспорту – це комбінаторна задача оптимізації і цілочисельного програмування, в якій для парку транспортних засобів, розташованих в одному або декількох депо, повинен бути визначений набір маршрутів до декількох віддалених точок – споживачів. Ця задача узагальнює відому задачу комівояжера.

Вперше поняття ЗМТ з'являється в статті [1]. Автори даної статті в 1959 році зробили математичне формулювання, а також вирішили задачу про постачання бензину на заправні станції. На сьогоднішній день ЗМТ є однією з найбільш відомих в області комбінаторної оптимізації. Слід зазначити, що вона відноситься до класу NP – повних задач, що означає, що її обчислювальна складність експоненціально залежить від розміру вхідних даних.

ЗМТ задача лежить на перетині двох добре вивчених задач:

1. Задача комівояжера. Якщо вантажопідйомність кожного транспортного засобу приймається як нескінченна (точніше, достатня), то ЗМТ зводиться до множинної задачі комівояжера (Multiple Traveling Salesman Problem, MTSP) шляхом додавання до вихідного графу $k - 1$ (де k – кількість маршрутів) копій нульових вершин та її ребер (між самими копіями ребра відсутні).

2. Задача пакування рюкзака. Вирішення даної задачі є еквівалентним вирішенню ЗМТ за умови, що всі відстані приймаються рівними нулю.

Залежно від заданих обмежень, характеристик вузлів та транспортних засобів ЗМТ можна класифікувати так:

1. ЗМТ з обмеженням по вантажопідйомності (ЗМТОВ). Це проблема МТЗ в якій парк ідентичних транспортних засобів з обмеженою

вантажопідйомністю, котрі розташовані в центральному депо, повинні обслуговувати набір клієнтів з відомими вимогами.

2. ЗМТ з часовими вузлами. Це проблема МТЗ, в якій обслуговування кожного із замовників повинне бути здійсненим у визначений проміжок часу.

3. ЗМТ зі зворотнім зв'язком. Це проблема МТЗ, в якій можливе повернення товару після завершення обслуговування всіх клієнтів (при чому кожна транспортна одиниця спочатку розвозить товари і лише потім починає приймати товари від клієнтів).

4. Періодична ЗМТ. У цій задачі вантаж може бути доставлений протягом періоду більше 1 дня.

5. Стохастична ЗМТ. Задача, у якій з ціллю опису дій покупця чи автотранспортних засобів можуть бути використані випадкові змінні.

6. ЗМТ із декількома типами транспортних засобів.

7. ЗМТ з підбором та доставкою.

1.2 Задача маршрутизації транспортного засобу із підбором та доставкою

Задача маршрутизації з можливістю підбору та доставки товарів відрізняється від стандартної задачі можливістю транспортного засобу забрати товар із одного вузла та перевезти в інший.

Постановка задачі включає в себе такі елементи, як вузли замовлень, термінальні вузли, а також транспортні засоби.

Вважатимемо, що загальна кількість замовлень n . Кожне замовлення $order_i$ характеризується вузлом підбору $pickUp_i$ та вузлом доставки $delivery_{i+n}$. Тоді, множина підбору $P = \{1, \dots, n\}$, а множина доставок $D = \{n + 1, \dots, 2n\}$. Нехай $N = P \cup D$. Позначимо множину всіх транспортних K , $|K| = m$ та множину термінальних вузлів T , $|T| = 2m$.

Позначимо початковий та кінцевий термінальний вузли для ТЗ k :

$$t_k = 2n + k, k \in K$$

$$t_k' = 2n + k + m, k \in K$$

Важливо також вказати, що термінальні вузли і вузли замовлень можуть співпадати між собою, тобто може бути один термінальний вузол для всіх ТЗ, або ж деякі з вузлів можуть бути як вузлами доставки, так і підбору. Маємо граф $G = (V, A)$ з вузлами $V = N \cup \{t_1, \dots, t_m\} \cup \{t_1', \dots, t_m'\}$ та дугами $A = V \times V$. Визначимо для кожної дуги відстань $distance_{ij} \geq 0$, а також тривалість подорожі $time_{ij} \geq 0$. Необхідно аби значення часу подорожей та відстаней задовольняли нерівностям трикутника:

$$distance_{ij} \leq distance_{il} + distance_{lj}, \quad \forall i, j, l \in V$$

$$time_{ij} \leq time_{il} + time_{lj}, \quad \forall i, j, l \in V$$

Також кожен вузол $i \in V$ характеризується часом обслуговування s_i та часовим вікном $[a_i, b_i]$, $a_i \leq b_i$. Під часом обслуговування маємо на увазі той час, котрий необхідно витратити на завантаження та розвантаження ТЗ у вузлі. Часове вікно визначає часові межі для обслуговування вузла. Кожен вузол $i \in P$ характеризується кількістю вантажу q_i , котрий необхідно забрати. Кожен вузол $j \in D$ характеризується кількістю вантажу q_j , котрий потрібно привезти у цей вузол. Якщо вузли належать до одного замовлення $j = i + n$, то $q_i = q_j$. Кожен вузол доставки/підбору відвідується лише раз. Також необхідно аби вузли підбору передували вузлам доставки у сформованих маршрутах.

Позначимо місткість транспортного засобу $k \in K$ як C_k . Для кожного транспортного засобу також маємо вартості α_k – вартість за одиницю шляху, β_k – вартість за одиницю часу, γ_k – вартість використання транспортного засобу.

Цільова функція може бути різною, а саме можлива мінімізація відстані та/або часу, вартості або кількості маршрутів, кількості замовлень, що не вдається обслужити, однак, загалом має наступний вигляд:

$$\min \sum_{k \in K} \alpha_k \sum_{(i,j) \in A} distance_{ij} X_{ijk} + \sum_{k \in K} \beta_k (\theta_{t'_k, k} - a_{t_k}) + \sum_{k \in K} \gamma_k y_k + \delta \sum_{i \in P} z_i,$$

де $\theta_{t'_k, k}$ – час прибуття ТЗ k у термінал t'_k ,

δ – штраф за один нерозподілений вузол,

X_{ijk} – булева змінна, котра рівна 1 у випадку, якщо дуга $(i, j) \in A$

обслуговується ТЗ $k \in K$,

y_k – булева змінна, котра рівна 1 у випадку використання ТЗ $k \in K$ у остаточному рішенні,

z_i – булева змінна котра рівна 1 у випадку, якщо замовлення i не було розподілено.

1.3 Аналіз актуальності задачі розміщення мобільних транспортних засобів

Зі світовою тенденцією зростання кількості міського населення перед місцевою владою виникає низка проблем: високий рівень руху, споживання енергії, шумове забруднення, атмосферне забруднення, тощо. У вирішенні таких проблем значно допомагають ССВМТЗ.

Першою ССВМТЗ був велосипедний пункт, встановлений в Амстердамі в 1965 р. За наступні роки кількість систем кардинально зросла досягнувши в 2011 р. понад 400 одиниць лише в Європі, детальніше у [2], [3]. Загалом, у всьому світі кількість таких систем зростає досить високими темпами (рис. 1.1), детальні описи та приклади можна знайти у [4].

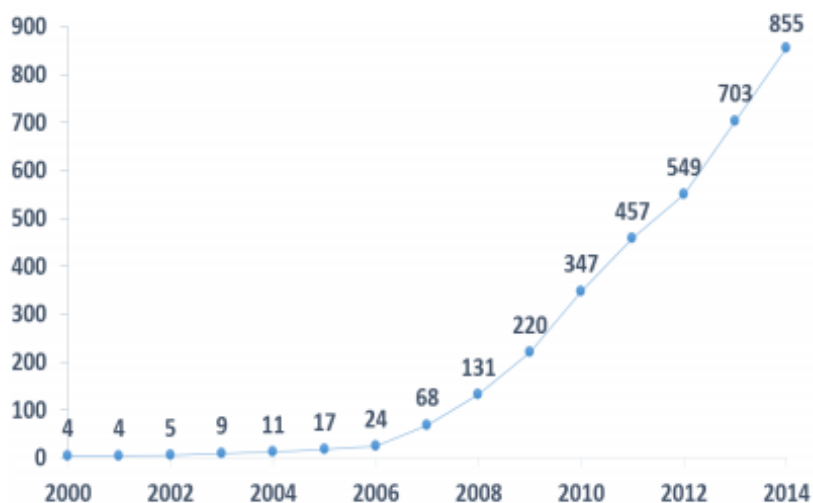


Рисунок 1.1 – Кількість міст у Європі, що використовують системи обміну мобільними транспортними засобами

За останні роки ССВМТЗ стали надзвичайно розповсюдженими у великих містах та агломераціях, їх популярність постійно зростає. Сьогодні вони вже стали невід'ємною частиною міського середовища. Їх вибирає як молодь, так і старше покоління, тому що навчитися користуватися ними можливо всього за пару хвилин. А з урахуванням переваг даного засобу пересування майже всі хотіли б користуватися таким видом транспорту кожен день.

Більш того, врахувуючи події що відбуваються в сучасному світі, суспільство повинне звернути увагу на проблеми екології та навколишнього середовища. Автомобілі, мотоцикли, моторолери та частина суспільного транспорту щодня заповнюють вулиці міста, серйозно забруднюючи природу. Наслідки цього – не тільки забруднення повітря, але й зайвий шум, через який проживання в центрі міста стає все менш придатним для життя.

ССВМТЗ є надзвичайно корисними для інфраструктури кожного міста та є важливим інструментом, що використовується державними адміністраціями для отримання більш стійкої мобільності, зменшення інтенсивності дорожнього руху, усунення завантаження, а також екологічного й шумового забруднення, спричинених транспортними перевезеннями.

Також ССВМТЗ відіграють важливу роль у вирішенні проблеми «останньої милі», під якою розуміється переміщення людей чи вантажів від транспортного вузла до пункту кінцевого призначення.

Наразі системи байкшернігу належать до тих, котрі мають найбільшу швидкість зростання у світі із загальним ростом близько 45%, починаючи з 2007. В кінці 2014 року сукупно було близько 800 міст, в котрих було встановлено системи байкшерінгу (рис. 1.2).

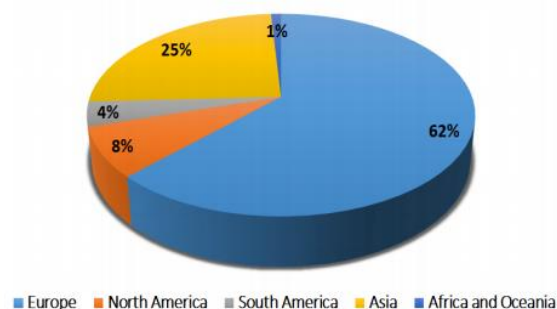


Рисунок 1.2 – Розповсюдження систем байкшерінгу у світі

Нещодавно, під час досліджень [5], також було виявлено, що ССВМТЗ допомагають місцевій владі значно зекономити на дорожніх затратах – якості доріг, обслуговуванні, тощо. Таким чином, відомо, що завдяки таким системам у Вашингтоні, окрузі Колумбія, кожен учасник дорожнього руху економить близько 800 долларів, в той час як кожна поїздка приносить близько 7 долларів місцевій економіці.

Зокрема починаючи з 2018 р. у містах України, а саме у Києві, Львові, Харкові, Одесі та Вінниці також були впроваджені байкшерінг системи, тому задача перерозподілу мобільних транспортних засобів є актуальною і для нашої країни загалом.

1.4 Опис проблеми ребалансування мобільних транспортних засобів

ССВМТЗ пропонують послуги, в яких громадськості надається транспорт для швидкого пересування доступний для спільного користування. Зазвичай, цей транспорт знаходиться на станціях, які розташовуються по місту за певними закономірностями. Користувачі системи можуть взяти ТЗ зі станції, використати його для свого переміщення та залишити на будь – якій іншій станції, що розташована у місті (не обов'язково станції відправлення), оплата відбувається відповідно до часу використання.

У світі існує досить багато таких систем, однак найбільш розповсюдженими серед населення є велосипеди, тому надалі під ССВМТЗ вважатимуться системи байкшерінгу. Зазвичай станція може мати вигляд локації із різними слотами в кожному з яких розміщений один велосипед або ж загального пустого приміщення без слотів. У сучасних системах станції підключені до Інтернету, тому легко отримати актуальні дані про ситуацію завантаженості кожної із станцій. Програмне забезпечення для оновлення даних працює у реальному часі, а зібрана інформація використовується для підвищення рівня обслуговування.

Експлуатація ССВМТЗ має вартість, що може сильно варіюватися (залежно від самої системи, щільності населення, площі обслуговування, розміру ринку, тощо). Витрати на встановлення системи включають вартість придбання ТЗ, слотів і станцій, тощо. Щоденні експлуатаційні витрати включають обслуговування, страхування, розробку та підтримку веб – сайтів, а також електроенергію. Однак, на сьогоднішній день головною проблемою лишаються витрати на перерозподіл велосипедів між станціями. Проблема полягає у тому, що наприкінці дня деякі станції, як правило, заповнені, а інші порожні. Першопочатковим та загальноприйнятим правилом підтримки достатньої кількості транспорту на кожній станції є лишити кожна з них лише частково зайнятою(якщо вона має обмежену слотами кількість місць). Мається на увазі, що на станції завжди має бути кілька слотів, зайнятих велосипедами, щоб хоча

б частково задовольнити попит користувачів, а також декілька вільних слотів, котрі б дозволили користувачам лишати велосипеди в кінці своєї подорожі. Поширеним підходом також є розподіл загальної кількості ТЗ у системі порівну між станціями аби лишити станції напівпустими/напівповними.

Зрозуміло, що бажана кількість ТЗ присутня рано вранці, за відкриття на певній велосипедній станції, однак кількість велосипедів може різко змінитися протягом дня з бажаного рівня до нестачі чи перевантаження через поведінку подорожуючих користувачів. Особливо цей дисбаланс виявляється в містах, що характеризуються горбистою територією [6], де користувачі беруть велосипед на станції котра розташована на вершині пагорба, лишують його внизу, а потім вирушають у зворотній шлях використовуючи зовсім інший вид транспорту.

На сьогодні, однією з основних проблем в управлінні ССВМТЗ є забезпечення можливості оренди та паркування велосипедів на будь – якій станції, в будь – який час. Для вирішення цієї проблеми застосовується ребалансування. Під цим терміном розуміється перепозиціонування товару за допомогою ємних транспортних засобів на базі центрального депо. Наприклад, вантажівки чи автобуси забирають велосипеди зі станцій, де попит занадто низький та доставляють їх до станцій, де він високий. Також, зазвичай певна кількість додаткових велосипедів зберігається в депо і використовується для дозволу більш широкого розподілу, а також заміни деяких ТЗ у разі поломки. Ребалансування – дуже дорога операція з точки зору логістики, її неефективність є однією з основних причин невдоволення клієнтів. Тобто, перед компаніями постає проблема перерозподілу велосипедів, відома в літературі як проблема ребалансування велосипедів (Bike Rebalancing Problem), що являє собою проблему оптимізації про те, як максимально задовольнити потреби користувача, передбачити попит на конкретних станціях та прокласти шлях для транспортних засобів таким чином, аби здійснити ребалансування системи з мінімальними затратами.

Слід також зазначити, що проблема ребалансування ССВМТЗ включає в себе проблему ЗМТ із підбором та доставкою.

1.5 Аналіз існуючих підходів до вирішення проблеми ребалансування мобільних транспортних засобів

1.5.1 Стратегічне планування

В початкових дослідженнях задачі ребалансування, в багатьох роботах особлива увага зверталася на розміщення байкшерніг станцій таким чином, аби забезпечити максимальну ефективність їх роботи та мінімальну необхідність перерозміщення. Місця для локацій та кількість станцій визначалась за допомогою різних факторів, зокрема приватних даних місцевих жителів.

Так автори в [7] використовують лінійну регресійну модель для вивчення кореляції між розташуванням станцій та клієнтів, котра має такий вигляд:

$$Y_{locality} = \beta_{INTERCEPT} + \beta_{ME} \times X_{ME, locality} + \beta_{PT} \times X_{PT, locality} + \beta_{CR} \times X_{CR, locality} + \beta_{CC} \times X_{CC, locality} + \beta_{HOT} \times X_{HOT, locality} + \beta_{CL} \times X_{CL, locality} + \beta_{Dist} \times X_{Dist,locality} + \beta_{GT} \times X_{GT,locality} + \varepsilon,$$

де $Y_{locality}$ – продуктивність однієї із станцій в конкретній локації,

$\beta_{INTERCEPT}$ – зміщення,

β_{ME} – коефіцієнт змінної, котра відповідає за наявність менеджерів та експертів,

$X_{ME, locality}$ – змінна, що відповідає за кількість менеджерів та експертів у конкретній локації,

β_{PT} – коефіцієнт змінної, котра характеризує пасажиропотік на громадському транспорті,

$X_{PT, locality}$ – змінна, що відповідає за кількості випадків використання громадського транспорту у конкретній локації,

β_{CR} – коефіцієнт змінної, котра відповідає за поїздки на особистому транспорті,

$X_{CR, locality}$ – змінна, що відповідає за кількість випадків використання особистого транспорту у конкретній локації,

β_{CC} – коефіцієнт змінної, що відповідає за наявність торгових центрів,

$X_{CC, locality}$ – змінна що відповідає за наявність торгових центрів, характеризується кількістю кроків від торгового центру до конкретної локації,

β_{HOT} – коефіцієнт змінної, що відповідає за наявність готелів чи хостелів (тимчасового житла),

$X_{HOT, locality}$ – змінна, що відповідає за наявність готелів чи хостелів, характеризується кількістю кімнат в хостелах та готелях для конкретної локації,

β_{CL} – коефіцієнт змінної, що відповідає за наявність навчальних закладів,

$X_{CL, locality}$ – змінна, що відповідає за наявність навчальних закладів, характеризується кількістю студентів в навчальних закладах конкретної локації,

β_{Dist} – коефіцієнт змінної, що відповідає за наявність інших байкшерінг станцій,

$X_{Dist,locality}$ – кількості інших байкшерінг станцій в даній локації, на відстані від неї не більше 500 метрів,

β_{GT} – коефіцієнт змінної, що відповідає за активність використання послуг даної компанії,

$X_{GT,locality}$ – змінна, що відповідає за активність використання послуг даної компанії,

ε – похибка.

Продуктивність k – ої станції представимо у такому вигляді:

$$\begin{aligned} \text{Exp}(Y_k) = & \beta_{INTERCEPT} + \beta_{ME} \times X_{ME, k} + \beta_{PT} \times X_{PT, k} + \beta_{CR} \times X_{CR, k} + \\ & + \beta_{CC} X_{CC,k} + \beta_{HOT} \times X_{HOT, k} + \beta_{CL} \times X_{CL, k} + \beta_{Dist} \times X_{Dist,k} \end{aligned}$$

Тоді математична модель матиме вигляд:

$$\text{Max} \sum_{k=1}^K z_k \text{Exp}(Y_k) \quad (1.1)$$

$$\sum_{k=1}^K z_k = n \quad (1.2)$$

$$X_{Dist,k} = \sum_{k'} z_{k'} \quad \forall k \quad (1.3)$$

$$X_{PT,k} = \sum_{s \in S_k} \left(\frac{r_s}{\sum_{k' \in k_s} z_{k'}} \right) \quad \forall k \quad (1.4)$$

$$n \leq |K|, \quad (1.5)$$

де K – множина усіх можливих локацій,

k – множина потенційно вигідних локацій,

$s \in S$ – множина усіх зупинок громадського транспорту,

r_s – кількість поїздок від кожної із них,

S_k – множина зупинок, котрі знаходяться в області впливу потенційної станції $s_k \subset S$,

k_s – множина локацій, на котрі має вплив станція s , $k_s \subset K$,

z_k – двійкова змінна, котра рівна 1 у випадку якщо k вибрали у якості станції.

Основною ціллю моделі є забезпечення компромісу між перспективною кількістю клієнтів у конкурентній локації та густоти станцій у цьому ж місці. Зокрема у згаданій роботі стверджується, що занадто багато станцій у місцях високого попиту не збільшать загальну продуктивність системи. Дана модель допомагає у виявленні привабливих локацій, котрі могли б лишитись непоміченими.

Формулювання (1.1) - (1.5) є задачею нелінійної цілочисельної оптимізації, котру автори вирішують за допомогою евристичного підходу. Таким чином при вирішенні задачі спочатку знаходять найкращий набір Y_s на основі

усіх параметрів, окрім відстані та громадського транспорту. На першій ітерації, вважаємо що усі станції є такими, що підходять для вигідних локацій, тобто $z_k = 1 \forall k$ і за допомогою цього знання обчислюємо X_{PT} . Надалі, вибираємо найкращі n локацій на основі цільової функції. Опісля, значення змінної X_{PT} перераховується, враховуємо що вибрано лише n конкретних локацій. Процес повторюється ітеративно до моменту того, коли набір станцій стане сталим.

Основним недоліком моделі є те, що відстань між станціями недооцінюється. Так, наприклад, автори пропонують звернути увагу на введення додаткових станцій в центрі, однак робити це з особливою обережністю. Також основним недоліком є те, що модель значно залежить від особистих даних користувачів, а також врахований дуже вузький набір факторів, котрі впливають на попит. Так, наприклад, можна ще врахувати вплив цін на громадський транспорт, наявність парковочних місць, тощо.

Цікавими є також роботи [8], [9], де автори пропонують вирішувати дану проблему за допомогою моделі підтримки прийняття рішень. Зокрема, вони враховують частково необхідність обслуговування станцій, звертаючись до літератури пов'язаної з управлінням запасами.

Загалом, задача планування ефективного розміщення є досить важливою для ССВМТЗ. Однак, лише оптимальне розміщення не зможе вирішити проблему нерівномірного попиту та пропозиції на місцях. Відповідно до багатьох зовнішніх чинників (популярність того чи іншого району для роботи/житла, наявність громадського транспорту, тощо) дуже важко знайти такий спосіб розміщення станцій аби кількість транспортних засобів до кінця дня на місцях залишалась відносно сталою і не було необхідності у ребалансуванні, тому все ж таки, на сьогодні, основна увага сконцентрована на вирішенні задачі перепозиціонування.

1.5.2 Статичне ребалансування

Більшість станцій у світі наразі використовують статичне ребалансування. Під цим терміном розуміють таке ребалансування, при якому в період перерозподілу велосипедів рух між станціями є незначним. Тобто, основним завданням є пошук маршрутів для фіксованого набору транспортних засобів для переміщення велосипедів увечері, вночі або ж вкінці робочого дня, коли переміщення ТЗ клієнтами є незначним, для досягнення бажаної конфігурації на базових станціях.

Існує досить багато робіт, котрі розглядають проблему перерозміщення лише за наявності одного автомобіля. Так, у роботі [10] пропонується приблизне рішення на основі задачі комівояжера з прийомом та доставкою одного товару, використовуючи лише один автомобіль для ребалансування. Кожну вершину можна відвідувати по декілька разів, а також використовувати у якості буфера.

Пізніше, у праці [11] автори розглядають проблему як транспортну задачу доставки та прийому товару одного типу з обмеженням на пропускну здатність. Автори також використовують один автомобіль для перерозподілу велосипедів. Вирішення проблеми здійснюється за допомогою методу гілок та меж, також використовується пошук із заборноюю. Почтакове рішення, необхідне для табу – пошуку знаходиться за допомогою жадібного методу або методу релаксації. Представлений у задачі підхід дозволив вирішувати досить об'ємні задачі з кількістю станцій більше 100.

Однак перерозподіл за допомогою єдиного ТЗ є ефективним лише для станцій невеликого розміру та малих відстаней. Для великих систем байкшерінгу, зазвичай, використовують декілька транспортних одиниць, так як одного автомобіля недостатньо для об'їзду усіх станцій. Більш того, перебалансування зазвичай займає досить багато часу, а цей ресурс є обмеженим.

Знову ж таки, підходи, що використовують декілька ТЗ розглянуті у [10], [11] не є ефективними, оскільки знаходження маршруту для першого ТЗ керує

рішенням маршрутизації інших транспортних засобів і впливає на загальну якість рішення.

У роботі [12], [13] автори враховують необхідність використання декількох вантажівок та пропонують масштабовані підходи для знаходження точного та наближеного рішення, формулюють проблему як задачу дискретної оптимізації. Зокрема у другій роботі автори пропонують дві моделі. А саме індексовану по часу, а також по ребрам між наявними пунктами.

Особливістю обох запропонованих рішень є наявність цільової функції, котра має дві складові. Перша – частина із штрафними функціями, котрі «штрафують» за відсутність вільного місця для велосипедів або ж за наявності порожніх слотів на довгий проміжок часу, друга – експлуатаційні витрати. Зокрема, автори пропонують використання ваг для виокремлення окремих видів недостач та їх мінімізації, так як кожна недостача по різному впливає на рівень задоволення клієнта. Автори надають приклади бенчмарків для обох формулювань. Загалом, індексована по часу модель дає кращі результати за обмеження CPU часу до 2 годин. Однак, множина допустимих рішень для індексованої по ребрам моделі є меншою. Це спричинено додатковим обмеженням, а саме тим, що кожна станція може бути відвідана кожною вантажівкою лише 1 раз. Загалом, більш точні результати були отримані для моделі з індексацією по ребрам, не зважаючи на те, що обмеження для цієї моделі є більш строгими. Індексована по часу модель втрачає точність через можливість синхронізації та прямого зв'язку з іншими вантажівками. Така постановка задачі більше підходить для динамічного варіанту ребалансування. Основна різниця у представленні двох формулювань наведена у табл. 1.1.

Таблиця 1.1 – Порівняння можливостей та припущень, щодо ребалансування для індексованих по часу та ребрам формулювань

Припущення	Індексація по ребрам	Індексація по часу	Сценарій допустимий при індексації по часу однак недопустимий при індексації по ребрам
Вантажівкам дозволено відвідувати кожну станцію необмежену кількість разів	Ні. Вантажівка може відвідати станцію лише один раз. Станція може бути відвідана декількома вантажівками	Так	Вантажівка їде до станції 1, розвантажує певну кількість велосипедів, їде до 2 – ої станції, завантажує певну кількість велосипедів, повертається до станції 1 аби знову вивантажити їх
Перевантаження дозволені, а отже вантажівки можуть розвантажувати велосипеди у деяких пунктах задля того, щоб потім більше завантажитись у інших пунктах	Так, однак максимальна кількість розвантажених велосипедів не може перевищувати початковий рівень велосипедів на станції	Так	Спочатку станція пуста. В деякий момент часу, вантажівка розвантажує 15 велосипедів на станції, а пізніше інша вантажівка забирає 5 велосипедів на цій станції (котрі раніше не були тут наявні)
Дозволені простої на пунктах для синхронізації перевантажень	Ні, перевантаження може бути реалізоване лише у вигляді вивантаження і завантаження на станціях	Так	Вантажівка приїжджає на станцію, чекає близько 5 хв на іншу вантажівку й забирає певну кількість велосипедів вивантажену іншою вантажівкою
Всі велосипеди повинні бути вивантажені, перед тим, як вантажівка повернеться до депо	Так	Ні	Вантажівка приїжджає у депо із 10 велосипедами, забирає додаткові 5 і продовжує свій шлях

Зовсім інший підхід до вирішення задачі статичного ребалансування використали автори [14], [15], [16] котрі пропонують вирішувати задачу за допомогою локального пошуку зі змінними околами.

У роботі [17] автори формулюють задачу з використанням інтервалів попиту для кожного автомобіля і вирішують її, як задачу цілочисельного програмування. Зокрема для цієї задачі пропонується використання схеми декомпозиції Бендерса для ефективного вирішення проблеми.

Загалом, усі підходи згадані вище є унікальними. Вони застосовуються лише для статичної задачі ребалансування та не можуть бути використані у динамічному підході. Основними недоліками статичного ребалансування є те, що воно не працюватиме у випадках, коли система байкшерінгу активно працює уночі. Так як умови для обслуговування байкшерінг станцій міняються з часом через мимовільне активне переміщення клієнтів, статичні рішення можуть швидко дати збій, у випадку активного попиту чи пропозиції. Динамічне ребалансування дає змогу врахувати поточну ситуацію на станції та приймати рішення в реальному часі.

1.5.3 Динамічне ребалансування

У динамічних версіях ребалансування відбувається моніторинг системи в режимі реального часу, план перерозподілу оновлюється, як тільки з'являється інформація необхідна для прийняття рішень (рис. 1.3).

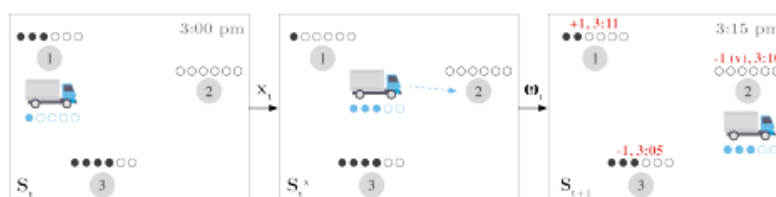


Рисунок 1.3 – Візуалізація проблеми динамічного ребалансування

У роботі [18] автори пропонують динамічне репозиціонування, котре направлене на подолання втрат споживачів у час пік. Вони забезпечують динамічність рішення, враховуючи нещодавній попит, котрий спостерігався до прийняття рішення про репозиціонування. Крім того, автори використовують методи декомпозиції Данцига Вульфа та Бендерса для прискорення процесу вирішення задачі. Однак, через велику складність поставленої задачі, знайдене рішення дещо далеке від оптимального. Крім того, беручи до уваги, що дане рішення враховує лише короткочасні потреби, воно зазвичай не є підходящим у випадку, коли попит змінюється динамічно.

У роботі [19] автори використовують дещо відмінний підхід, котрий вирішує згадану вище проблему за допомогою розрахунку очікуваного попиту на весь проміжок часу роботи станції. Передбачення майбутнього попиту здійснюється за допомогою розподілу Пуассона на основі історичних даних із метро про поїздки користувачів. Саме за допомогою цих даних робляться висновки про активність користувачів у той чи інший час доби. Автори пропонують оптимізаційну модель, котра знаходить найкращі локації для розміщення станцій, а також модель динамічного репозиціонування, котра направлена на мінімізацію кількості незадоволених клієнтів. Однак, автори не розглядають саме прокладання маршруту, вони вважають, що ребалансування є завжди можливим у довільному порядку та однаково ефективним.

Новий та цікавий підхід пропонують автори у роботі [20]. Вони поєднують балансування за допомогою вантажівок, а також взаємодією із користувачами. Автори використовують гейміфікацію, а саме спонукають користувачів виправити систему, впливаючи на їх поведінку за допомогою винагород і призів (рис. 1.4). Додатковий стимул надається споживачам, якщо вони повертають свої велосипеди до сусідньої станції, котра має недостачу велосипедів, а не відвозять їх на початкову станцію призначення. Щодо самого ребалансування, автори пропонують динамічне переміщення та маршрутизацію для окремих ТЗ на основі оцінки попиту протягом наступних 30 хвилин. У разі наявності багатьох автомобілів, використовується жадібний підхід, де рішення для кожного із ТЗ визначаються послідовно по одному.

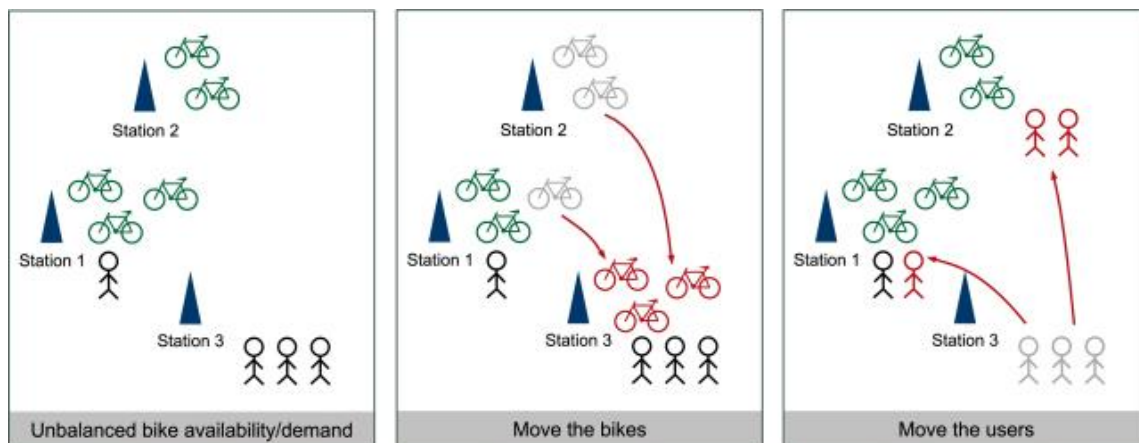


Рисунок 1.4 – Динамічне ребалансування із впливом на користувачів системи

Використовуючи аналогічний підхід, автори [21] пропонують механізм ціноутворення, який динамічно формує цінове заохочення для користувачів, за умови доставлення велосипедів до заданої станції та відповідно пропонує певні суми користувачам через мобільний додаток. За допомогою заохочення користувачів автори вирішують лише проблеми недостачі велосипедів на конкретних станціях. Загалом робота поділяється на 4 етапи:

1. Визначення оптимальної кількості велосипедів для кожної станції.
2. Врахування змін попиту протягом часу.
3. Знаходження для користувача підходящої станції для оренди/повернення, котра знаходилась би якомога найближче до його поточного місцезнаходження.
4. Рорахунок бонусу для клієнта в рамках обмеженого бюджету.

У даній роботі заохочення для користувачів направлене на підтримання балансу на станціях. Так, для ситуації на рис. 1.5 користувача, котрий бажає взяти велосипед та знаходиться найближче до пункту S_1 необхідно заохотити для здійснення оренди в пункті S_2 , так як для першої станції більш необхідною є операція повернення, ніж оренди.

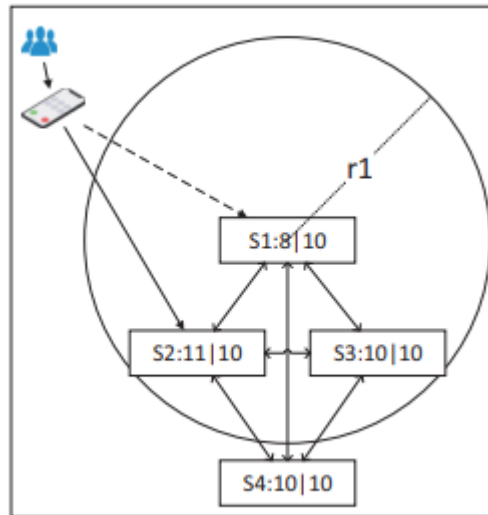


Рисунок 1.5 – Приклад динамічного ребалансування при взаємодії із користувачами

Також вдалим є використання Марківського процесу прийняття рішень у роботі [22] для вирішення стохастичної проблеми маршрутизації для ССВМТЗ. Ціль полягає у тому, аби мінімізувати кількість очікуваних порушень. Під порушенням тут розуміють крайній термін, до котрого станція повинна бути відвідана автомобілем, котрий виконує перепозиціонування, щоб задовольнити попит та уникнути збоїв, а також втрати клієнтів. Саме у цій статті показана важливість використання, як довгострокових так і короткострокових стратегій планування. Використання довгострокових стратегій використовується для передбачення майбутнього попиту, а короткострокові визначають пріоритет ребалансування для кожної станції, керуючись тим, аби мінімізувати втрату клієнтів.

1.5.4 Прогнозування попиту на станціях для ребалансування

Для вирішення багатьох задач статичного та динамічного ребалансування активно використовується прогнозування майбутнього попиту для кожної із станцій з метою їх забезпечення необхідною кількістю ТЗ. Для прогнозування найчастіше використовують наступні підходи:

1. Баєсова мережа – графічна модель, що представляє собою ациклічний орієнтований граф, вершинами якого виступають випадкові події, а дуги позначають існування безпосереднього причинно – наслідкового зв'язку між вершинами і орієнтовані від причини до слідства. Обчислення ймовірностей подій в мережі відбувається з використанням закону множення ймовірностей і теореми Баєса. До основних переваг методу належить те, що він дає хорошу точність прогнозування незважаючи на маленький об'єм вибірки. Основними недоліком є неможлива безпосередня обробка неперервних даних, а також погане масштабування, низька швидкість обчислення.

2. Екстра рандомізовані дерева. Вони відрізняються від звичайних дерев прийняття рішень тим, що розподіл робиться в рандомних точках, а також використовують всю навчальну вибірку для побудови дерева і таким чином ідентифікують закономірності, що присутні в цій вибірці (рис. 1.6).

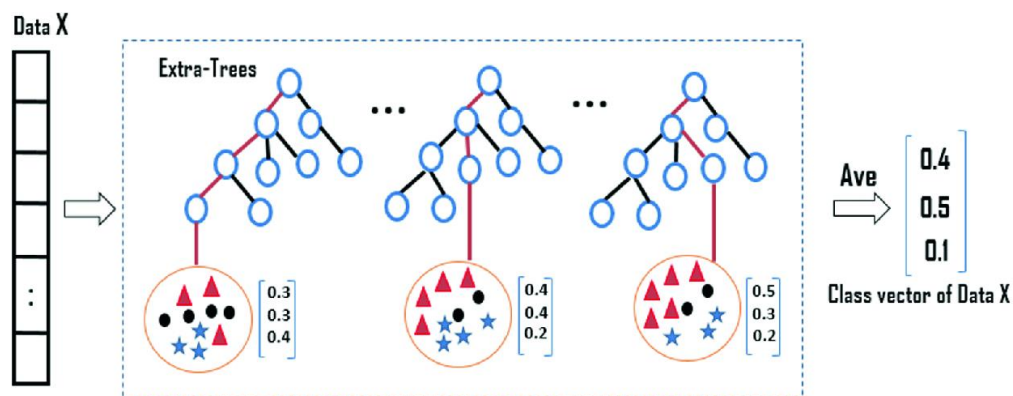


Рисунок 1.6 – Застосування Екстра рандомізованих дерев

До основних переваг можна віднести високу швидкість обчислення та легку паралелізацію. До основних недоліків відноситься потреба у великій кількості пам'яті для зберігання моделі.

3. Градієнт бустинг – це техніка машинного навчання для задач класифікації і регресії, яка будує модель передбачення в формі ансамблю слабких прогнозуючих моделей, зазвичай дерев рішень, котрі послідовно мінімізують помилку один одного і в результаті, в комбінації добре описують навчальну вибірку вхідних даних (рис. 1.7). Основними перевагами є надійність, продуктивність, зменшення обсягу часу необхідного для обробки вхідних даних. Перенавчання можна уникнути за допомогою правильного вибору параметрів. Також градієнт бустинг має хорошу точність прогнозування.



Рисунок 1.7 – Приклад роботи градієнт бустингу

Автори [23] у своїй роботі здійснили порівняння результатів 3 підходів, а саме градієнт бустингу, лінійної регресії, а також нейронних мереж. Градієнтний бустинг показав найкращі результати для вибірки даних байкшерінг систем Бостону. Важливо також зазначити що він має перевагу, котра є надзвичайно важливою для роботи в реальному часі, а саме модель автоматично перенавчається в міру отримання нових даних. Середня похибка необхідного наповнення станції за використання градієнт бустингу становила 3.6%, 9.5%, 11.2% для часових проміжків 20, 40 та 60 хвилин.

4. Лінійна регресія. Найпростішим варіантом регресійної моделі є лінійна регресія. В основу моделі покладено припущення, що існує дискретний

зовнішній фактор $X(t)$, який впливає на досліджуваний процес $Z(t)$, при цьому зв'язок між процесом і зовнішнім фактором лінійний. Модель прогнозування на підставі лінійної регресії описується рівнянням:

$$Z(t) = \alpha_0 + \alpha_1 X(t) + \varepsilon t,$$

де α_0 і α_1 – коефіцієнти регресії,

εt – помилка моделі.

До основних переваг відносять швидкість та простоту моделі, точні результати у випадку лінійної залежності. До недоліків те, що її неможливо застосувати для вирішення нелінійних задач. Також, слід зазначити, що у роботі [23] цей підхід показав гірші результати за градієнт бустинг.

5. Нейронні мережі. Перевага нейронних мереж полягає в здатності моделювати нелінійні зв'язки без попередніх припущень про структуру даних. Для завдання регресії на останньому шарі нейронної мережі завжди повинна бути використана функція активації що не змінює області допустимих значень для прогнозу. Найчастіше використовуються лінійна функція активації і випрямлена лінійна функція активації (Rectified linear unit, ReLU). До основних недоліків підходу належить те, що реалізація є досить ресурсоємною. Також, слід зазначити, що у роботі [23] цей підхід показав гірші результати за градієнт бустинг.

6. Регресія Пуассона. У статистиці регресія Пуассона є узагальненою лінійною моделлю регресійного аналізу, що використовується для моделювання даних з непередбачених ситуацій. Регресія Пуассона передбачає, що залежна змінна Y має розподіл Пуассона, і припускає, що логарифм її математичного сподівання може бути змодельований лінійною комбінацією невідомих параметрів. Переваги цієї моделі в тому, що вона має дискретний розподіл і має обмеження результатів до невід'ємних чисел. Також, у 2013 році було проведено дослідження компанією Data Science for Social Good у партнерстві з Департаментом транспорту Чикаго, з використанням регресії

Пуассона для прогнозування кількості клієнтів для конкретних локацій систем байкшерінгу. Результати досліджень показали, що цей підхід є досить легким, зручним та перспективним.

1.6 Висновки

У цьому розділі була розглянута ЗМТ, а також зроблений детальний огляд ЗМТ із підбором та доставкою. Було здійснено загальний огляд проблеми ребалансування для систем ССВМТЗ. Була проаналізована актуальність даної задачі, а також виявлено проблеми та можливі напрямки досліджень в цій галузі.

Основна увага приділялась підходам до вирішення задачі ребалансування, а саме стратегічному плануванню, статичному та динамічному ребалансуванню, розглянуті основні переваги та недоліки даних підходів. Перший полягає у знаходженні локацій для станцій МТЗ з умовою забезпечення подальшого балансу транспорту на місцях упродовж дня. Інші два розглядають проблему перерозподілу ТЗ між станціями у певний період часу таким чином, аби мінізувати затрати на перерозподіл та максимально задовольнити потреби користувача. Статичний метод здійснює ребалансування в період мінімального використання транспорту користувачами, тобто зазвичай вночі або увечері, а динамічний діє у реальному часі, тобто впродовж дня.

Окрім того, у розділі були представлені основні підходи до вирішення задачі прогнозування попиту на станціях, а саме Баєсівські мережі, лінійна регресія, нейронні мережі, екстра рандомізовані дерева, регресія Пуассона, градієнтний бустинг.

Подана у даному розділі інформація є основою для розуміння подальшого матеріалу, а саме застосування статичного підходу до вирішення задачі ребалансування для ССВМТЗ.

РОЗДІЛ 2 МЕТОД СТАТИЧНОГО РЕБАЛАНСУВАННЯ ДЛЯ СИСТЕМ СПІЛЬНОГО ВИКОРИСТАННЯ МОБІЛЬНИХ ТРАНСПОРТНИХ ЗАСОБІВ

2.1 Задача статичного ребалансування мобільних транспортних засобів

Задача статичного ребалансування мобільних транспортних засобів полягає у визначенні найвигіднішого маршруту для обходу станцій зі збором або постачанням транспортних засобів парком автомобілів із обмеженою місткістю, за часу мінімальної активності користувачів системи.

Постановка задачі має наступний вигляд. Задано набір вершин $V = \{0, 1, \dots, n\}$, де 0 - ва вершина це депо, а множина вершин $\{1, \dots, n\}$ це станції. Для кожної із станції наданий попит - кількість необхідних одиниць транспорту q_i , котрий може бути як додатнім, так і від'ємним, попит у депо є нульовим $q_0 = 0$. Додатній попит $q_i > 0$ означає, що i – вузол, з котрого необхідно забрати велосипеди зі станції (маємо перевантаження на станції), від'ємний попит $q_i < 0$ означає що i – вузол, у котрий потрібно доставити велосипеди (маємо недостачу на станції). Велосипеди, котрі були вилучені з вузлів можуть прямувати до інших вузлів доставки або ж в депо. Велосипеди, що постачаються на станції можуть бути взяті як з депо, так і зі станції із надмірною кількістю транспорту. На складі наявно m потенційно різних транспортних засобів із ємністю Q , котрі відповідають за перепозиціонування транспорту між станціями. Також задана відстань подорожі c_{ij} із вершини i у вершину j .

Основною ціллю є визначення такого маршруту для m транспортних засобів, щоб затрати були мінімальні та виконувались наступні обмеження:

1. Маршрут кожного транспортного засобу починається та закінчується у депо.
2. Кожен транспорт вирушає із депо порожнім або з деяким навантаженням $p_j < Q, j = \overline{1, m}$.

3. Сума попитів відвіданих станцій разом із початковим навантаженням ніколи не є від'ємною або більшою за Q на маршруті, який виконує транспортний засіб.

4. Кожна станція відвідується лише раз.

Важливим завданням для вирішення задачі ребалансування є також обробка та аналіз вхідних даних. Зазвичай у якості вхідних даних використовуються локації станцій, тривалість поїздки користувача та її маршрут, статистика по конкретним станціям, а саме кількість наявних велосипедів у певні часові проміжки, кількість необслужених користувачів, кількість непридатних для використання велосипедів, тощо (рис.2.1). Саме правильний аналіз та інтерпретація даних визначає необхідні умови та особливості побудови оптимізаційної моделі для знаходження маршруту.

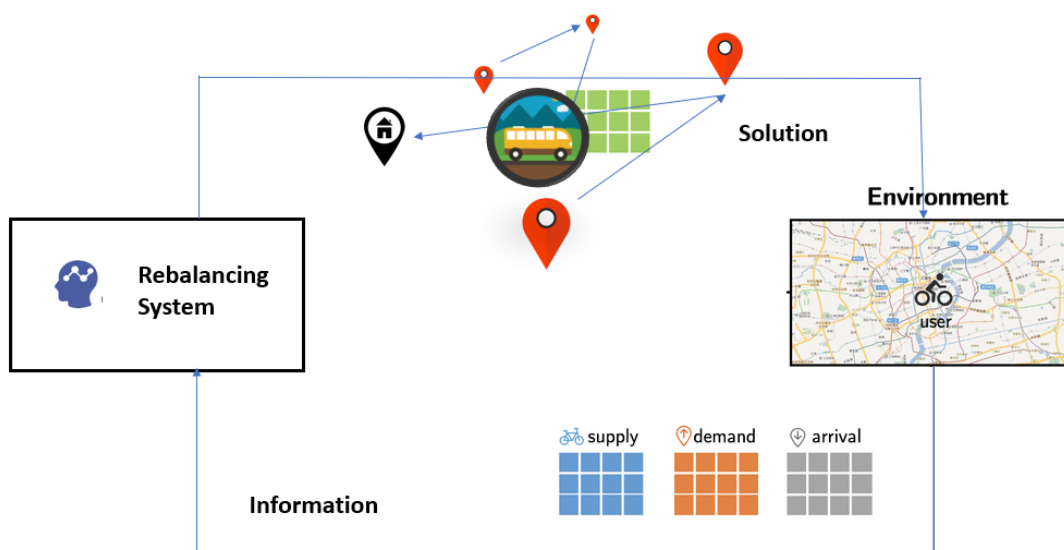


Рисунок 2.1 – Система для вирішення проблеми ребалансування

2.2 Прогнозування попиту

2.2.1 Визначення патернів поведінки станцій

Більшість існуючих методів прогнозування короткострокового попиту на станціях засновані на історичній моделі змін кількості ТЗ на станціях. З плином часу, після встановлення у міському районі системи прокату МТЗ, у жителів розвивається звичка до стаціонарних поїздок. Ці звички є відносно стабільними у будні дні та дещо відрізняються у вихідні.

На кожній із станцій прокату велосипедів буде спостерігатися періодична схожість. Іншими словами, на велосипедних станціях існуватимуть фіксовані вхідні та вихідні патерни (рис. 2.2).

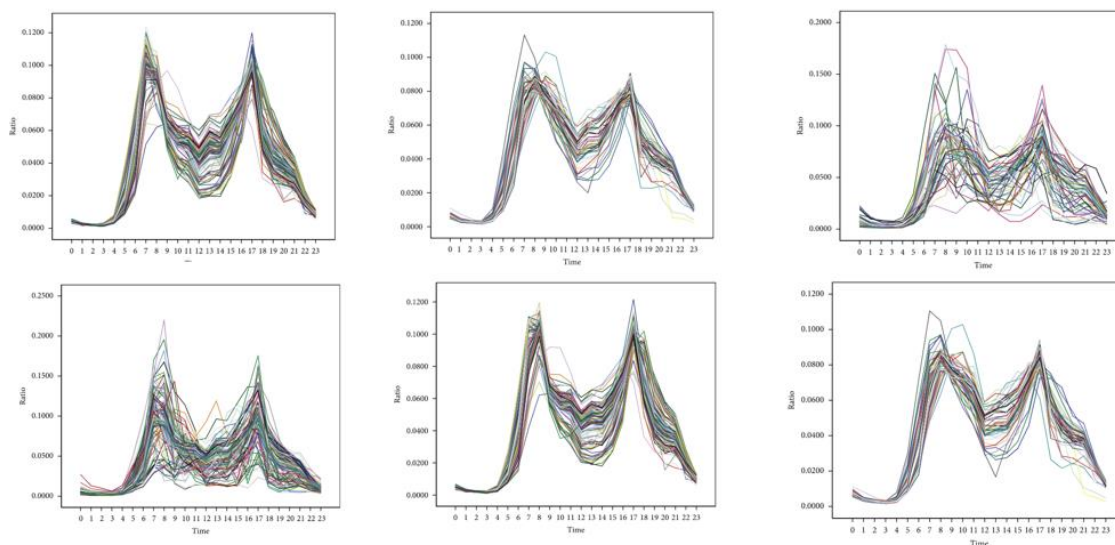


Рисунок 2.2 – Поведінкові патерни 6 станцій у Бостоні терміном 4 місяці

Саме ці патерни і використовуються для визначення типів станцій.

Для типізації станцій використовують значення чистого потоку ТЗ на станції:

$$H_t = Inflow_t - Outflow_t,$$

де H_t – чистий потік,

$Inflow_t$ – кількість вхідних для станції ТЗ за одиницю часу t ,

$Outflow_t$ – кількість вихідних для станції ТЗ за одиницю часу t .

У результаті, станції зазвичай поділяють на станції постачальники, прийому, а також постачальники – прийому. Також існує певний набір станцій, у яких попит та пропозиція часто співпадають, значення H_t у таких станцій рівне 0, або ж може коливатись на невеликі значення. Зазвичай, саме ці станції користувачі використовують для вирішення проблеми «останньої милі» у повсякденному житті.

Належність станції до конкретного виду значно залежить від типу місцевості та близькості до центру міста, а також наявності зупинок громадського транспорту. Для першого та третього типу станцій попит на транспортні засоби визначається як сума поточної кількості ТЗ та мінімального значення чистого потоку, для станцій прийому попит являє собою суму поточної кількості ТЗ із різницею велосипедів вкінці та на початку дня.

Безпосередньо корисною є побудова кривих чистого потоку. Вони використовуються для аналізу динаміки попиту, відображають різницю ТЗ за певний проміжок часу. Для станцій отримувачів кількість вихідних велосипедів завжди більша за кількість вхідних, тому така станція потребує для поставки певну кількість ТЗ наприкінці дня. Для станцій постачальників, кількість вихідних велосипедів значо менша за кількість вхідних, тому така станція потребує збору певної кількості ТЗ, для третього типу значення чистого потоку може значно коливатись і приймати як від’ємне так і додатне значення. Приклади кривих чистого потоку наведені на рис. 2.3.

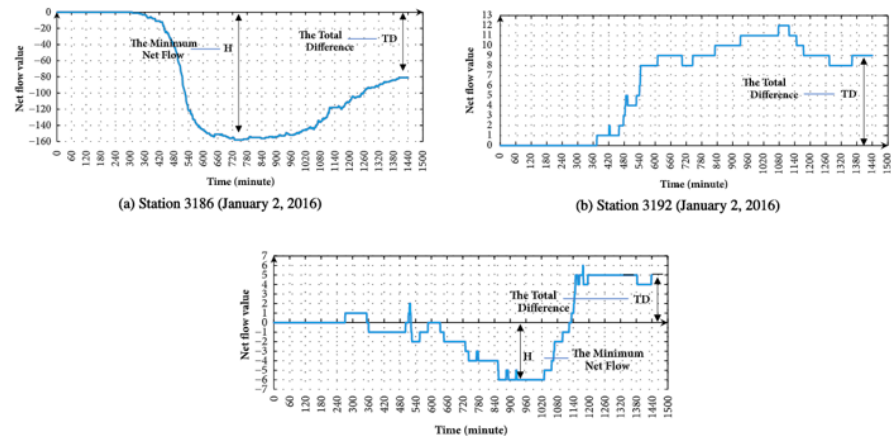


Рисунок 2.3 – Приклади кривих чистого потоку для різних станцій: а) станція отримувач; б) станція постачальник; в) станція отримувач – постачальник

2.2.2 Визначення попиту

Попит станції на ТЗ це кількість ТЗ, що вона потребує для доставки або збору. Розглянемо деяку i – ту станцію, котра є станцією отримувачем або ж постачальником – отримувачем. Як показано на рис. 2.4, попит i – ої станції за день t визначається мінімальним значенням чистого потоку $H_{i,t}$. Позначимо різницю транспортних засобів на початку та вкінці дня на i – й станції, як $SD_{i,t}$. Змінна $SD_{i,t}$ відображає дисбаланс між попитом та пропозицією. Вважатимемо, що $N_{i,t}$ – кількість велосипедів наприкінці попереднього дня ($t - 1$) на i – й станції.

Тоді загальний попит i – ої станції $q_{i,t}$ за день t визначатиметься так:

$$q_{i,t} = N_{i,t} + H_{i,t}, \quad (2.1)$$

де $q_{i,t}$ – попит на станції i для t – го дня,

$N_{i,t}$ – кількість велоспедів на i – й станції наприкінці дня ($t - 1$),

$H_{i,t}$ – мінімальне значення чистого потоку транспортних засобів за день t .

Для станції постачальника ситуація буде наступною. $H_{j,t}$ за день t буде рівним 0, а $SD_{j,t} > 0$. Тоді попит на транспортні засоби на цій станції визначатиметься так:

$$q_{j,t} = N_{j,t} + SD_{j,t}, \quad (2.2)$$

де $q_{j,t}$ – попит на станції j для t – го дня,

$N_{j,t}$ – кількість велоспедів на j – й станції вкінці дня ($t - 1$),

$SD_{j,t}$ – різниця між кількістю велосипедів на j – й станції вкінці та на початку t – го дня.

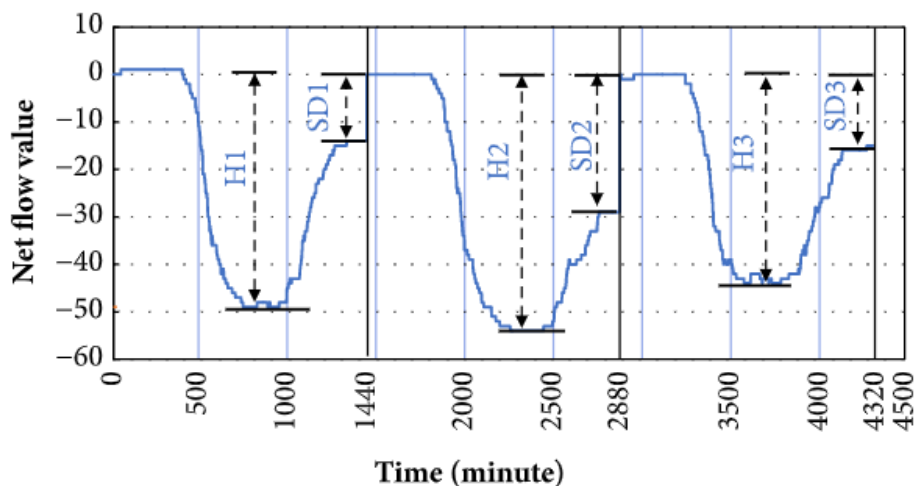


Рисунок 2.4 – Крива чистого потоку для станції отримувача

Для визначення недостачі транспортних засобів у системі спільного використання транспорту достатньо обрахувати попит для станцій. Загальний попит для системи за певний період t обраховується наступним чином:

$$TD = \sum_i q_{i,t}$$

У випадку $TD < 0$ – додаткова кількість велосипедів із депо повинна бути введена у систему, у випадку $TD > 0$ – певна кількості велосипедів повинна бути вилучена із системи для її продуктивного функціонування. Вилучення непотрібних ТЗ зазвичай здійснюється в час перепозиціонування, а саме лишня кількість ТЗ прямує до депо.

2.2.3 Прогнозування максмального щоденного попиту станції

У (2.1) та (2.2) $N_{i,t}$ та $N_{j,t}$ задані, а значення $H_{j,t}$ та $SD_{j,t}$ необхідно спрогнозувати. Існує багато методів для роботи із часовими рядами такі як множинна лінійна регресія, експоненціальне згладжування, штучні нейронні мережі, тощо. Серед заданих методів одним із найбільш зручних для прогнозування є градієнтний бустинг.

Градієнтний бустинг заснований на постійному ітеративному зважуванні і корекції спостережень у вибірці при постійному відслідковуванні результативності при проведенні операції. Операції корекції повинні зменшувати помилку, зроблену на попередніх кроках. Модель для прогнозування будується у вигляді ансамблю слабких прогнозуючих моделей (рис.2.5).

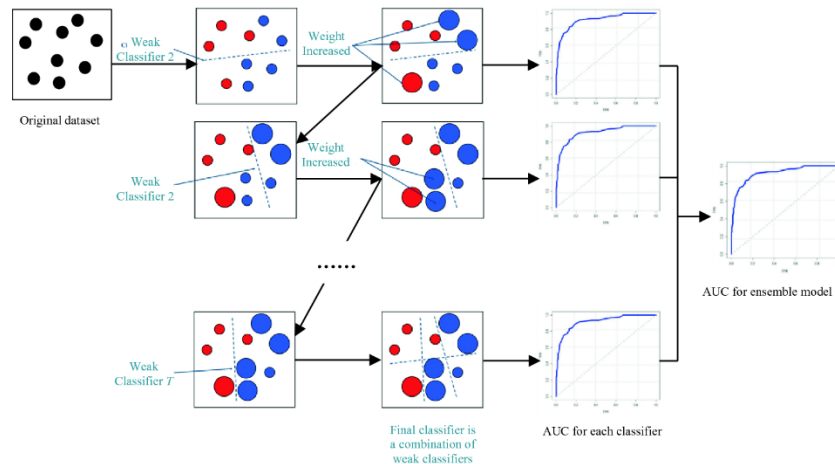


Рисунок 2.5 – Приклад роботи градієнтного бустингу

Нехай задана тренувальна вибірка $Z = \{(x_1, y_1), \dots, (x_n, y_n)\}$, x_i – вектор незалежних змінних, y_i – результуюча змінна. Основною ціллю є знаходження $f(x)$, використовуючи тренувальну вибірку таку, щоб

$$\min \sum_{(x,y) \in T} (f(x) - y)^2,$$

де $T = \{(x_1, y_1), \dots, (x_n, y_n)\}$.

У поточному випадку матимемо:

$$f(x) = \sum_{m=1}^M h_m(x),$$

де $h_m(x)$ – дерево рішень.

Побудова дерева відбувається за допомогою жадібного підходу зверху донизу.

Розподіл дерев відбувається за допомогою теоретико – інформаційного критерію, що базується на максимізації величини отриманої інформації IG :

$$IG = \text{Error before split} - \text{Error after split}$$

Розглянемо детальніше як відбувається побудова. Нехай задана вибірка $Z = \{(x_1, y_1), \dots, (x_n, y_n)\}$. Ціллю є знаходження такого дерева $f(x)$, щоб $\min \sum_{i=1}^n (f(x_i) - y_i)^2$.

На кожному кроці треба знайти найкращий розподіл у вузлі. Нехай є деяка вибірка Z і критерій розділу $x_k < t$ (рис. 2.6).

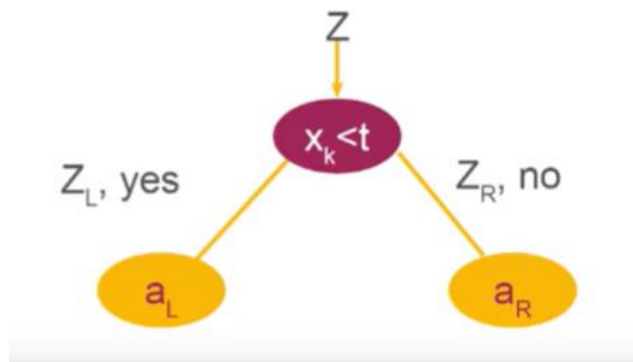


Рисунок 2.6 – Розділ початкового вузла

Необхідно знайти поріг t та номер ознаки k , а також значення листків дерева a_L та a_R . Розглянемо ситуацію без розподілу. У цьому випадку можливий прогноз значення a . Доцільно вибрати таке \hat{a} аби мінімізувати середньоквадратичну помилку:

$$\hat{a} = \min_a \sum_{i \in Z} (a - y_i)^2$$

Виберемо \hat{a} таким чином:

$$\hat{a} = \frac{1}{|Z|} \sum_{i \in Z} y_i,$$

де $|Z|$ – кількість елементів у Z .

Маємо помилку:

$$\text{Impurity}(Z) = \frac{1}{|Z|} \sum_{i \in Z} (\hat{a} - y_i)^2$$

У випадку розділу за критерієм $X_k < t$ матимемо дві множини Z_L у яку входять елементи, для яких виконується критерій розділу та Z_R , у яку входять елементи, для яких не виконується критерій розділу.

Ціллю є:

$$\min_{k,t,a_L,a_R} \sum_{i \in Z_L} (a_L - y_i)^2 + \sum_{i \in Z_R} (a_R - y_i)^2$$

Вважатимемо, що :

$$\hat{a}_L = \frac{1}{|Z_L|} \sum_{i \in Z_L} y_i,$$

де $|Z_L|$ – кількість елементів у Z_L .

$$\hat{a}_R = \frac{1}{|Z_R|} \sum_{i \in Z_R} y_i,$$

де $|Z_R|$ – кількість елементів у Z_R .

Тоді матимемо:

$$\min_{k,t} \sum_{i \in Z_L} (\widehat{a}_L - y_i)^2 + \sum_{i \in Z_R} (\widehat{a}_R - y_i)^2$$

Необхідно знайти t та k . Маємо, що:

$$Impurity(Z_L) = \frac{1}{|Z_L|} \sum_{i \in Z_L} (\widehat{a}_L - y_i)^2,$$

$$Impurity(Z_R) = \frac{1}{|Z_R|} \sum_{i \in Z_R} (\widehat{a}_R - y_i)^2$$

Отже, можемо знайти найрацій критерій розподілу котрий максимізує IG у відношенні до k, t (критерій поділу $x_k < t$):

$$IG = Impurity(Z) - \left(\frac{|Z_L|}{|Z|} Impurity(Z_L) - \frac{|Z_R|}{|Z|} Impurity(Z_R) \right)$$

Процедура є ітеративною. В якості критерію зупинки виберемо обмеження глибини дерева. Сам метод градієнт бустингу працює як градієнтний спуск, однак у функціональному просторі.

2.2.4 Виключення збалансованих станцій

Варто зазначити, що після прогнозування, а також обчислення попиту для конкретних станцій, деякі із станцій є збалансованими, а саме їхній попит рівний 0, $q_i = 0$. Така ситуація найчастіше трапляється зі станціями, котрі знаходяться біля метро або автобусних зупинок. Ними зазвичай користуються люди, що живуть у віддалених районах міста і, власне, використовують ТЗ ситеми лише

для того, аби доїхати до зупинки. Такі станції відразу виключаються із множини станцій для ребалансування.

2.3 Математична модель задачі перерозподілу транспортних засобів

2.3.1 Початкова математична модель

При побудові моделі вважатимемо, що:

- Кількість транспортних засобів, котрі здійснюють ребалансування більше 1 ($m > 1$);

- Кількість товарів для доставки або ж підбору є спрогнозованою.

Тоді модель матиме наступний вигляд:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij} x_{ijk} \quad (2.3)$$

$$\sum_{i \in V} \sum_{k \in K} x_{ijk} = 1, \quad \forall j \in V_0 \quad (2.4)$$

$$\sum_{i \in V} x_{ihk} = \sum_{j \in V} x_{hjk}, \quad \forall h \in V_0; \forall k \in K \quad (2.5)$$

$$\sum_{j \in V_0} \sum_{k \in K} x_{0jk} \leq m, \quad (2.6)$$

$$\sum_{j \in V_0} x_{0jk} = \sum_{i \in V_0} x_{i0k}, \quad \forall k \in K \quad (2.7)$$

$$Q_{tot} = \sum_{j \in V_0} q_j, \quad (2.8)$$

$$L_{0k} = \max \left\{ 0, - \sum_{i \in V} \sum_{j \in V_0} q_j x_{ijk} \right\}, \quad \forall k \in K \quad (2.9)$$

$$\sum_{k \in K} L_{0,k} \geq \max\{0, -Q_{tot}\}, \quad (2.10)$$

$$L_{k,j} \geq L_{0k} + q_j - M(1 - x_{0jk}), \quad \forall k \in K, \forall j \in V_0 \quad (2.11)$$

$$L_{kj} \geq L_{ki} + q_j - M \left(1 - \sum_{k \in K} x_{ijk} \right), \forall i \in V_0, \forall j \in V_0, i \neq j \quad (2.12)$$

$$L_{0k} \leq Q, \quad \forall k \in K \quad (2.13)$$

$$L_{kj} \leq Q, \quad j \in V_0, k \in K \quad (2.14)$$

$$\sum_{i \in V_0} \sum_{j \in V_0} \sum_{k \in K} x_{ijk} \leq |S| - \left\lceil \frac{\max\{0, -Q_{tot}\}}{Q} \right\rceil, \quad (2.15)$$

$$x_{ijk} = \begin{cases} 1, & \text{якщо ребро } (i, j) \text{ використовувавалось } k \text{ - ою вантажівкою,} \\ 0, & \text{у іншому випадку,} \end{cases} \quad (2.16)$$

де V – множина вершин, де $V_0 \subseteq V \setminus \{0\}, S \neq \emptyset$,

A – множина ребер,

n – кількість станцій,

K – множина вантажівок,

k – номер вантажівки, $k = 1, 2, \dots, m$,

m – кількість вантажівок,

Q – вантажопідйомність вантажівки,

q_i – попит у i – й вершині,

c_{ij} – довжина подорожі ребром (i, j) ,

L_{0k} – початкове завантаження k – ої вантажівки,

L_{k0} – кінцеве завантаження k – ої вантажівки, коли та прямує до депо,

L_{kj} – завантаження k – ої вантажівки після вершини $j, j \in V_0$,

Q_{tot} – загальний попит,

x_{ijk} – двійкова змінна.

Цільова функція (2.3) мінімізує затрати на маршрут. Обмеження (2.4) відповідає за те, аби кожна станція була відвідана лише один раз однією вантажівкою. Обмеження (2.5) відповідає за те, що жодна із вантажівок не припиняла свій шлях у вершині h . (2.6) накладає обмеження на загальну кількість вантажівок, а (2.7) стверджує те, що всі вантажівки починають та

закінчують свій шлях у депо. (2.8) показує, що загальний попит системи рівний сумі попитів окремих вершин. (2.9) встановлює початкове завантаження k – ої вантажівки. Загальне завантаження усіх вантажівок з депо у будь-якому випадку повинне бути невід'ємним. Більш того, якщо загальний попит Q_{tot} є негативним, то воно не повинне бути меншим від нього, що забезпечує обмеження (2.10). (2.11) накладає обмеження на завантаження k – ої вантажівки після першої станції. (2.12) накладає обмеження на завантаження k – ої вантажівки після станції j . (2.13) та (2.14) накладають обмеження на максимальне завантаження вантажівки відповідно до вантажомісткості автомобіля. Обмеження (2.15) є класичне обмеження, що виключає підтури (рис. 2.7). За відсутності цього обмеження можливе отримання рішення, що містить вироджені маршрути між проміжними вузлами, котрі не пов'язані з депо.

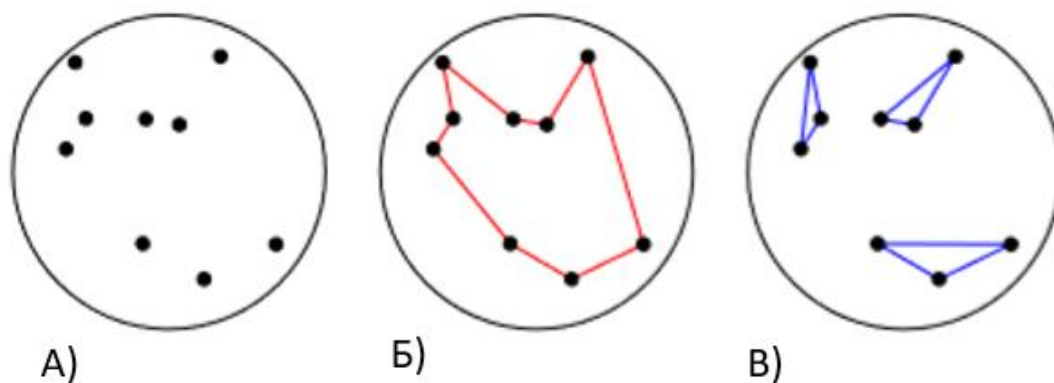


Рисунок 2.7 – Приклад рішення без/з врахуванням умови виключення підтурів: А) набір станцій; Б) рішення з урахуванням умови виключення підтурів; В) Рішення без врахування умови виключення підтурів

2.3.2 Врахування додаткових умов та обмежень

Важливим етапом для правильної побудови моделі є також врахування додаткових чинників, котрі можуть значно вплинути на стан ССВМТЗ та потягнути за собою необхідність змінити маршрут перепозиціонування. У даній роботі будуть розглянуті такі як врахування кількості пошкоджених ТЗ, а також характеристик конкретних автомобілів, що використовуються для перепозиціонування.

Після тривалого існування системи, в ній з'являється певний набір транспортних засобів котрі вже не здатні правильно функціонувати. Поява у ССВМТЗ ТЗ у аварійному стані призводить до порушення прогнозування попиту, а як наслідок, операції ребалансування загалом. Очевидно, що після тривалого періоду використання деякі велосипеди потребуватимуть ремонту. У випадках, коли велосипеди з поломками продовжують лишатись на станціях необмежений час, системний оператор може прийняти їх за експлуатаційні велосипеди. Більш того, при прогнозуванні попиту враховується кількість ТЗ, що лишилась на станції вкінці дня, в результаті кількість необхідних ТЗ наступного дня буде порахована невірно, що призведе до дефіциту ТЗ на станціях та більшого відсотку незадоволених запитів користувачів, а отже втрати коштів. Таким чином, важливою проблемою є виявлення таких засобів у аварійному стані та повернення їх у депо, що значно збільшує складність проблеми статичного ребалансування.

На відміну від традиційної проблеми маршрутизації транспортного засобу при одночасному підборі та доставці (VRPSPD), кількість зібраних велосипедів зі станцій, що потребують ремонту вже не може бути використана для задоволення від'ємного попиту. Оскільки очікується що пошкоджені ТЗ займуть певну частину місткості транспортного засобу, шлях авто для репозиціонування повинен бути заново спланованим.

Для вирішення описаної вище проблеми дещо змінимо побудовану модель та додамо деякі обмеження. Позначимо p_i – кількість пошкоджених ТЗ у вершині i , $p_i \geq 0$.

Тоді маємо:

$$\sum_{k \in K} L_{k0} \geq \sum_{j \in V_0} p_j, \quad (2.17)$$

$$L_{jk} \geq L_{0k} + q_j + p_j - M(1 - x_{0jk}), \quad \forall k \in K, \forall j \in V_0 \quad (2.18)$$

$$L_{kj} \geq L_{ki} + q_j + p_j - M \left(1 - \sum_{k \in K} x_{ijk} \right), \forall i \in V_0, \forall j \in V_0, i \neq j \quad (2.19)$$

$$\sum_{i \in V_0} \sum_{j \in V_0} \sum_{k \in K} x_{ijk} \leq |S| - \left[\frac{\max\{0, -Q_{tot}\} + \sum_{i \in V_0} p_i}{Q} \right] \quad (2.20)$$

Обмеження (2.17) вказує на те, що сумарна завантаженість усіх вантажівок котрі їдуть у депо більше (у випадку наявності лишніх ТЗ) або ж рівна кількості велосипедів у аварійному стані, тобто обмеження гарантує, що всі такі велосипеди були зібрані зі станцій. (2.18) – (2.20) – обмеження, що еквівалентні обмеженням (2.11), (2.12), (2.15) з урахуванням кількості пошкоджених ТЗ.

Отже (2.3) – (2.10), (2.13), (2.14), (2.16) – (2.20) кінцева математична модель для задачі перепозиціонування МТЗ із врахуванням пошкоджених ТЗ.

Слід також врахувати те, що попит на станціях із певною кількістю пошкоджених ТЗ дещо зміниться. Так, у випадку наявності пошкоджених ТЗ на станціях отримувачах, кількість ТЗ що необхідно довести збільшиться на величину пошкоджених ТЗ, для станцій постачальників кількість ТЗ, що необхідно забрати зменшиться на величину пошкоджених ТЗ.

Тобто:

$$q_i = q_{prev_i} - p_i,$$

де q_{prev_i} – попит станції i без врахування кількості поломаних ТЗ.

Надзвичайно важливим є також врахування характеристик конкретних автомобілів для здійснення ребалансування. Більшість байкшерінг компаній під витратами на ребалансування розуміють лише відстань, котру необхідно подолати аби зробити вдале репозиціонування для всіх станцій. Зазвичай компанії не звертають уваги на власний автопарк, котрий також значно впливає на витрачені кошти. При підрахунку затрат на перепозиціонування, важливим є також врахування ціни використання конкретного автомобіля, що здійснює цю операцію. При підборі вдалого автомобіля для перепозиціонування із врахуванням його технічного стану, кількості років експлуатації, а також витрат на пальне можливе значне зменшення витрачених коштів.

Позначимо Z_{auto} як коефіцієнт зносу транспортного засобу:

$$Z_{auto} = 1 - e^{-(\Delta T * T_{auto} + \Delta L * L_{auto})},$$

де ΔT – коефіцієнт, що визначає знос транспортного засобу відповідно до марки авто,

T_{auto} – термін експлуатації авто,

ΔL – коефіцієнт, що відображає зношення залежно від пробігу авто,

L – пробіг авто.

Ціновий коефіцієнт використання авто на км може бути відображений наступним чином:

$$Price_1 = (1 - Z_{auto}) * \frac{(Price_v)}{L_1},$$

де $Price_v$ – ринкова ціна автомобіля,

L_1 – пробіг автомобіля у км.

Позначимо $Price_2$ як середня ціна обслуговування автомобіля на км:

$$Price_2 = \frac{(Price_{rec})}{L_2},$$

$$L_2 = \frac{L}{T_{auto}},$$

де $Price_{rec}$ – середня ціна обслуговування автомобіля за рік,

L_2 – середній пробіг автомобіля за рік. У випадку експлуатації менше одного року – пробіг авто,

$Price_3$ – ціна використання пального на км для даного авто.

Матимемо наступну цільову функцію:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} (Price_{1_k} + Price_{2_k} + Price_{3_k}) c_{ij} x_{ijk} \quad (2.21)$$

Отже (2.21), (2.3) – (2.10), (2.13), (2.14), (2.16) – (2.20) кінцева математична модель для задачі ребалансування мобільних транспортних засобів із врахуванням особливостей парку автомобілів та кількості пошкоджених транспортних засобів.

2.4 Загальний опис методу статичного ребалансування

Нехай задана множина станцій для перепозиціонування $V = \{1, 2, \dots, n\}$.

Позначимо:

- $BikesInflow_i$ – кількість вхідних ТЗ для i – ої станції за проміжок часу t ;
- $BikesOutflow_i$ – кількість вихідних ТЗ для i – ої станції за проміжок часу t ;
- q_i – попит станції i ;
- p_i – кількість пошкоджених ТЗ на станції i ;
- $H_{i,t}$ – чистий потік на станції i за проміжок часу t ;
- $SD_{j,d}$ – різниця між кількістю велосипедів на початку та вкінці d – го дня на станції j ;
- H_{i, \min_d} – мінімальне значення чистого потоку за день d для станції i ;
- $Price_{1k}, Price_{2k}, Price_{3k}$ – цінові коефіцієнти для автомобіля k .

Отже, для знаходження кінцевого маршруту наступного дня для здійснення статичного ребалансування станцій повинні бути виконані наступні кроки:

1. Обрахунок $H_{i,t} = BikesInflow_{i,t} - BikesOutflow_{i,t}, i \in V$.
2. Розподіл станцій за типом. Віднесення станції i до певного типу здійснюється за значенням чистого потоку $H_{i,t}$. $H_{i,t} < 0 \forall t$ – станція отримувач, $H_{i,t} > 0 \forall t$ – станція постачальник. У випадку змінного значення $H_{i,t}$ станцію відносять до типу постачальник – отримувач.
3. Знаходження H_{i, \min_d} для станцій отримувачів та постачальників - отримувачів та $SD_{j,d}$ для станцій постачальників для $d \in \{1, \dots, n_1\}$, де n_1 – кількість днів в історичних даних необхідних для здійснення прогнозування.
4. Прогнозування H_{i, \min_d} та $SD_{j,d}$ для $d = n_1 + 1$.
5. Обрахунок q_i для кожної станції $i, i \in V$.

6. Вилучення станцій i , де $q_i = 0$ із множини станцій V для перепозиціонування:

7. Обрахунок $Price1_k, Price2_k, Price3_k$ для кожного автомобіля k автопарку, що здійснює перепозиціонування.

8. Модифікація значень q_i для врахування кількості пошкоджених ТЗ на станції i :

$$q_i = q_{prev_i} - p_i,$$

де q_{prev_i} – попит станції i без врахування кількості поломаних ТЗ,

p_i – кількість пошкоджених ТЗ на станції i .

9. Побудова математичної моделі. Вирішення задачі цілочисельного програмування.

2.5 Критерій якості ребалансування

Для оцінки якості прогнозування використовувався корінь середньоквадратичної помилки (RMSE) та середня абсолютна похибка (MAE):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (p_i - r_i)^2},$$

де n – кількість протестованих значень,

p_i – прогноз попиту на i – й станції,

r_i – реальне значення попиту на i – й станції.

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i|,$$

де n – кількість протестованих значень,

f_i – прогноз попиту на i – й станції,

y_i – реальне значення попиту на i – й станції.

Якість самого ребалансування оцінюється за допомогою відсотку задоволених запитів користувачів:

$$DE = 1 - \frac{N_{UD}}{N_{TD}}, \quad (2.22)$$

де DE – відсоток задоволеного загального попиту за день,

N_{UD} – кількість запитів користувачів, котрі неможливо було задовільнити за день,

N_{TD} – загальна кількість запитів за день.

2.6 Висновки

У даному розділі було детально розглянуто статичний метод ребалансування для ССВМТЗ. У цьому розділі була детально описана задача статичного ребалансування МТЗ та розроблено метод для її вирішення. Даний метод відрізняється від вже наявних тим, що враховує цінові характеристики авто та кількість пошкоджених ТЗ при проведенні операції перепозиціонування. Початковим етапом є аналіз даних та визначення патернів поведінки станцій. Загалом розрізняють три типи станцій – станції – постачальники, станції – отримувачі, а також отримувачі – постачальники. Визначення патернів станцій

та початковий аналіз даних є основою для знаходження подальшого попиту на станціях.

Дві головні складові методу – прогнозування попиту на станціях та побудова моделі для пошуку оптимального маршруту для перепозиціонування. Для здійснення прогнозування був вибраний метод градієнтного бустингу з деревами прийняття рішень. Також у розділі була представлена модель для перепозиціонування.

Побудована модель враховує наступні обмеження:

1. Маршрут кожного транспортного засобу починається та закінчується у депо.
2. Кожен транспорт вирушає із депо порожнім або з деяким навантаженням $p_j < Q, j = \overline{1, m}$.
3. Сума попитів відвіданих станцій разом із початковим навантаженням ніколи не є від'ємною або більшою за Q на маршруті, який виконує транспортний засіб.
4. Кожна станція відвідується лише раз.

У даному розділі також було наведено критерій якості ребалансування. Для оцінки якості прогнозування використовувалась корінь середньоквадратичної похибки та середня абсолютна похибка. Якість самого ребалансування оцінюється за допомогою відсотку задоволених запитів користувачів.

РОЗДІЛ 3 РОЗРОБКА СТРУКТУРИ ПРОГРАМНОГО КОМПЛЕКСУ

У даній роботі було розроблене програмне забезпечення для здійснення ребалансування ССВМТЗ. Програмний продукт надає користувачу можливість за заданою кількістю станцій та даними щодо щоденної активності кожної з них визначити тип станції, здійснити прогнозування попиту на кожній із станцій, а також визначити найкращий маршрут для здійснення операції перепозиціонування.

3.1 Обрані технології

Для розробки кросплатформного програмного забезпечення було обрано такі мови програмування як С++ та Python.

С++ є однією із найбільш популярних та поширених мов програмування на ринку розробки програмного забезпечення. Це об'єктно – орієнтована мова, котра поєднує низькорівневі та високорівневі можливості. Її основною перевагою є оптимальне використання обчислювальних ресурсів системи та можливість роботи із пам'яттю напряму. Написання коду на С++ надає можливість розробнику максимально адаптувати його до системних можливостей платформи, на котрій він працюватиме.

Python – об'єктно орієнтована мова програмування високого рівня, котра має строгу динамічну типізацію. Основною перевагою є кросплатформність. Python це інтерпретована мова, інтерпретатори існують для багатьох платформ, тому з запуском написаного коду не повинно виникнути проблем на будь – якій ОС. Також Python має величезну кількість доступних сервісів, бібліотек, середовищ розробки і фреймворків.

Для проведення попереднього аналізу даних, визначення типу, а також прогнозування та обрахунку попиту на кожній із станції використовувалась мова програмування Python, а також бібліотеки scikit – learn та numpy. Scikit – learn – бібліотека для аналізу даних, надає безліч можливостей для роботи із методами математичної статистики, штучних нейронних мереж, теорії нечітких множин або генетичних алгоритмів. numpy використовується цією бібліотекою для збереження датсетів у n – вимірних масивах, надає можливість роботи із великими багатовимірними масивами та матрицями, разом з великою бібліотекою високорівневих математичних функцій для операцій з цими об’єктами. Також використовувалась бібліотека Plotly для візуалізації даних та forecastio для визначення майбутніх та попередніх погодніх умов на станціях (температури, вологості, тощо).

Мова C++ використовувалась для здійснення самої операції перепозиціонування та знаходження оптимального шляху. Для знаходження рішення для заданої математичної моделі використовувався пакет програмного забезпечення CPLEX призначений для вирішення задач лінійного та квадратичного програмування, в тому числі цілочисельного програмування.

3.2 Структура програмного продукту

Реалізація програмного продукту вміщує у собі 2 основні частини: прогнозування попиту на станціях та знаходження оптимального шляху (рис. 3.1).

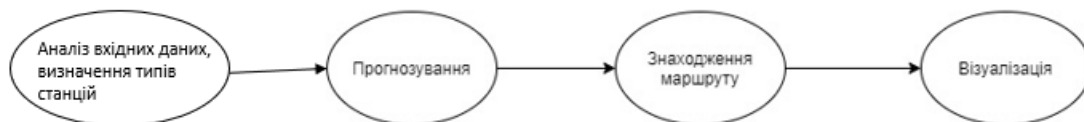


Рисунок 3.1 – Структура програмної реалізації

3.2.1 Структура системи прогнозування

На рис. 3.2 представлена архітектура системи прогнозування майбутнього попиту на одній із станцій.

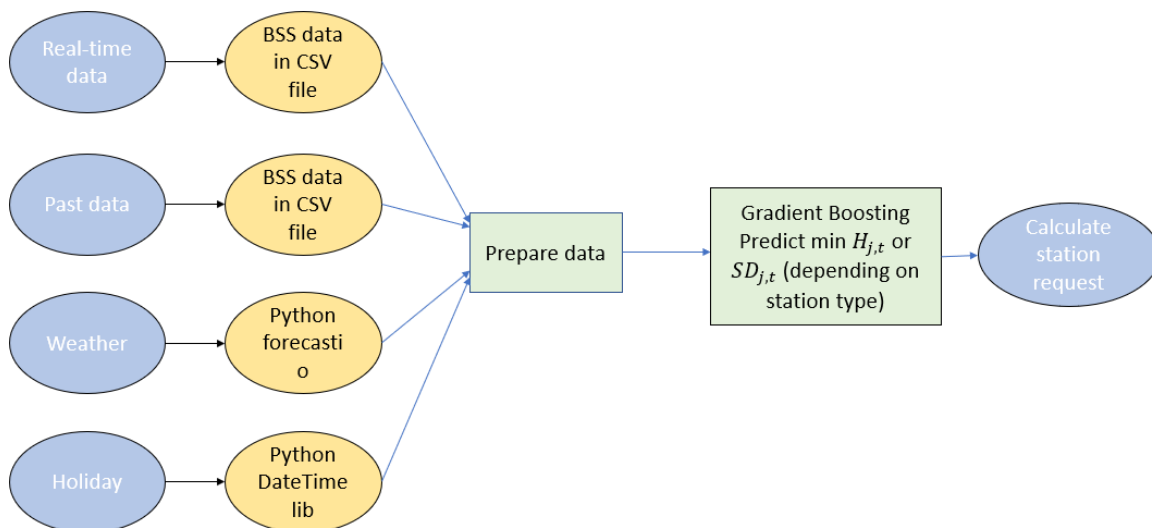
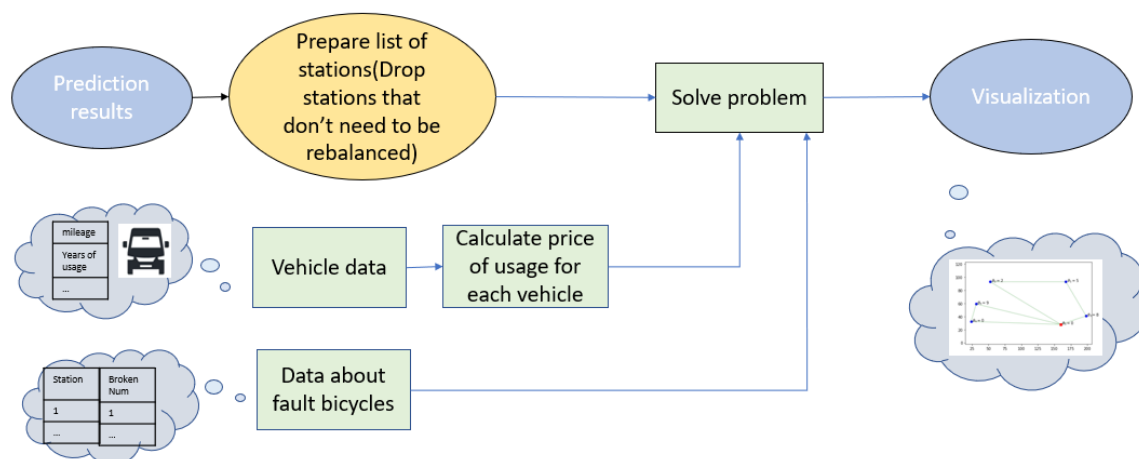


Рисунок 3.2 – Прогнозування майбутнього попиту для однієї із набору станцій

3.2.2 Структура системи перепозиціонування

На рис. 3.3 представлена архітектура системи для перепозиціонування ТЗ. Після отримання прогнозованого попиту, необхідно проаналізувати дані та відкинути станції із нульовим попитом задля покращення ефективності ребалансування. Також необхідно обрахувати попит, врахувавши певну к-ть пошкоджених ТЗ на станціях. Необхідним є наявність характеристик вантажівок, що використовуються для перепозиціонування для подальшого обчислення вартості використання кожної із них. Дані щодо пошкоджених ТЗ безпосередньо отримуються зі станцій.



Риснунок 3.3 – Структра системи ребалансування

3.3 Знаходження рішення задачі із врахуванням кількості пошкоджених транспортних засобів

3.3.1 Загальний огляд методу

Для вирішення заданої проблеми було застосовано метод чисельної оптимізації – дискретний метод рою часток (DPSO). Порівняно з генетичним алгоритмом, у нього немає параметрів ймовірності мутації та кросоверу; порівняно з алгоритмом імітації відпалу немає громіздкого процесу, що включає параметр, котрий відіграє роль температури; порівняно з мурашиним алгоритмом, немає різноманітних складних кроків судження та встановлення параметрів. Крім того, формула, що застосовується для оновлення швидкості та положення, використовується для забезпечення зближення всіх частинок до глобального оптимуму. Таким чином, дискретний метод рою часток має потужну можливість глобального пошуку. Однак у цього методу є наявні деякі недоліки. По – перше, встановлення різних параметрів матиме очевидний вплив на роботу DPSO. По – друге, алгоритм DPSO може легко сходиться до локального оптимуму. Отже, для подолання цих недоліків при збереженні переваг, у цій роботі було використано алгоритм гібридизації, що поєднує DPSO та пошук зі змінними околами, котрий був представлений у роботі [23]. Цей алгоритм може покращити здатність глобального пошуку DPSO та уникнути загрози лишитись в локальному оптимумі.

3.3.2 Представлення почтакового рішення

Початкове представлення рішення відіграє важливу роль. Правильно підібране початкове рішення може підвищити продуктивність алгоритму і знизити його складність. Загалом система містить N велосипедних станцій. Необхідно аби навантаження одного авто не перевищувало його місткість Q , а

кількість місць що лишилися у авто була б достатньою для переміщення/збору велосипедів для наступної станції та їх доставки до депо. Нехай кожна частинка являє собою вектор, що містить декілька маршрутів. Ініціалізуємо загальну кількість частинок додатнім цілим числом. У кожній частинці наявний розділяючий нуль, що має значення депо. Приклад частки, котра містить у собі три шляхи одного маршруту:

$$X = [1, 3, 6, 5, 0, 8, 11, 2, 4, 0, 7, 10, 9]$$

Перший шлях – 0 – 1 – 3 – 6 – 5 – 0, другий шлях – 0 – 8 – 11 – 2 – 4 – 0, третій шлях 0 – 7 – 10 – 9 – 0.

Вартість кожної із частинок рахується так:

$$cost = C_{0,X(i)} + \sum_{k=i+1}^{j-1} C_{X(k),X(k+1)} + C_{X(j),0}, \quad (3.1)$$

де $C_{j,j}$ – ціна проходження ребра (j, j) ,

$X(k)$ – відповідає за конкретну станцію, 0 – депо.

3.3.3 Зміна швидкості та позиції частинки

Зміна швидкості та позиції частинки відбувається за наступними правилами:

1. Швидкість. Швидкість частинки на ітерації t обчислюється наступним чином:

$$V_i^{t+1} = c_1 V_i^t + c_2 (P_i^t - X_i^t) + c_3 (G_i^t - X_i^t),$$

де V_i^{t+1} – швидкість частинки i на $t + 1$ ітерації,

V_i^t – швидкість частинки i на ітерації t ,

P_i^t – краще положення частинки i на ітерації t ,

G_i^t – глобальне найкраще положення усього рою,

c_1, c_2, c_3 – задані параметри.

2. Позиція. Позиція частинки на ітерації t оновлюється, враховуючи поточну позицію та швидкість частинки:

$$X_i^{t+1} = X_i^t + V_i^t$$

3.3.4 Застосування пошуку зі змінними околами

Для модифікації частинок застосовувались наступні процедури:

1. Вставка. Ця процедура полягає у тому, що випадкове число вставляється у випадкове положення частинки.

Після проведення цих процедур частинка обирається за наступним критерієм:

$$P_i^t = \begin{cases} P_i^{t,E}, & \text{якщо } f(P_i^{t,E}) < f(P_i^{t,0}), \\ P_i^{t,0}, & \text{якщо } f(P_i^{t,0}) < f(P_i^{t,E}), \end{cases}$$

де $P_i^{t,E}$ – частинка, що була змінена за допомогою процедури E на ітерації t ,

$P_i^{t,0}$ – частинка до виконання процедури на ітерації t ,

f – функція визначена у (3.1).

3.4 Знаходження рішення задачі з врахуванням характеристик автомобілів для ребалансування

Для знаходження рішення задачі з врахуванням характеристик автомобілів ребалансування використовувався метод гілок та відсіку. Для цього використовувався CPLEX 12.2 – пакет програмного забезпечення, що призначений для вирішення задач лінійного та квадратичного програмування, в тому числі цілочисельного програмування.

3.5 Висновки

У даному розділі було детально розглянуто процес створення програмного забезпечення для здійснення процедури статичного ребалансування. Було обгрунтовано вибір інструментів для реалізації та раціональність їх використання. Так, для знаходження шляху для перепозиціонування лише із

врахуванням ТЗ, що вийшли із ладу була обрана мова програмування C++ , основною перевагою якої є прикладний підхід до керування пам'яттю. Було реалізовано алгоритм гібридизації, що поєднує дискретний метод рою часток разом із локальним пошуком зі змінними околами. Для знаходження шляху перепозиціонування із врахуванням характеристик автомобілів та пошкоджених ТЗ використовувалась мова Python та пакет програмного забезпечення CPLEX 12.2, що призначений для вирішення задач лінійного та квадратичного програмування. Для прогнозування із використанням градієнтного бустингу застосовувалась мова Python із бібліотекою scikit – learn та numpy.

Значна увага приділялась структурі програмного забезпечення. Так прогнозування попиту станцій та знаходження маршруту для перепозиціонування ТЗ були створені як окремі модулі.

РОЗДІЛ 4 АНАЛІЗ ПРАКТИЧНОГО ДОСЛІДЖЕННЯ

4.1 Дані

Для перевірки роботи системи статичного ребалансування було обрано дані невеликого міста Арлінгтон, США із сумарною кількістю 11 станцій. Початковий датасет являє собою дані за кожною станцією впродовж 6 місяців починаючи із Серпня, 2017. Дані для станції містять дату, власне рік місяць та день, кількість вхідних та вихідних одиниць ТЗ впродовж роботи станції по годинно, кількість наявних велосипедів на станції, кількість користувачів котрим не вистачило транспортного засобу, тобто запити котрих не були задоволені. Також ці дані були доповнені типом дня (вихідний, будній день, свято), даними про погоду (температура, сонячно/пасмурно/туман/дощ) та порою року. На рис. 4.1 зображено приблизне розташування станцій. На рис. 4.2 зображено вигляд початкових даних.

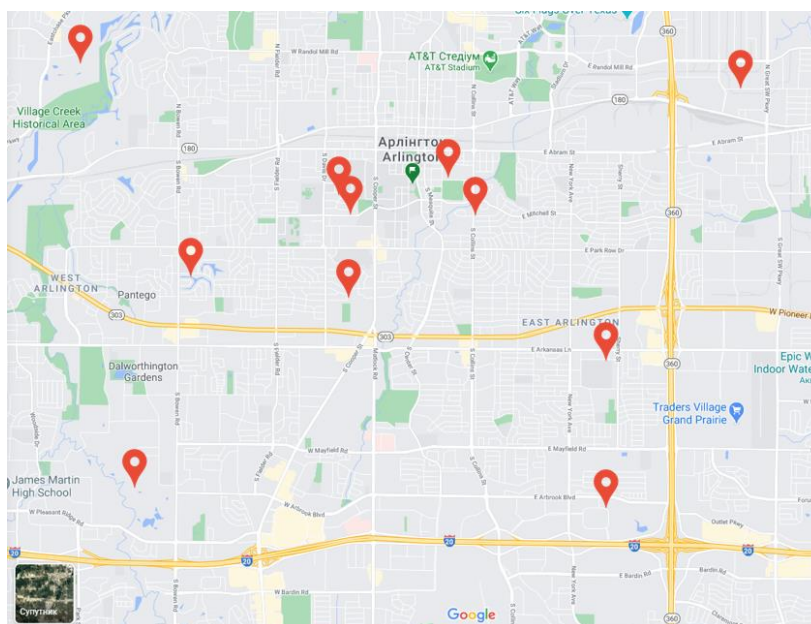


Рисунок 4.1 – Розташування станцій у місті

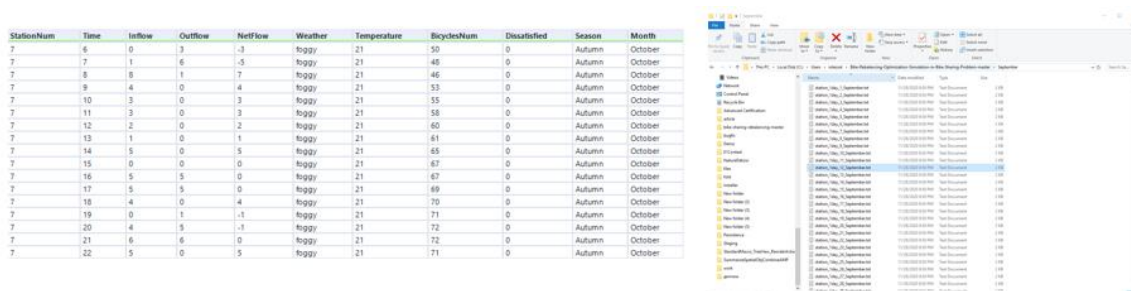


Рисунок 4.2 – Вигляд почтакових даних

Після аналізу чистого потоку станцій було виявлено 4 станції постачальника, котрі знаходяться ближче до центру міста та великих офісних будівель, 3 станції постачальника – отримувача, котрі знаходяться на невеликій відстані від автобусних зупинок та метро, а також 4 станції отримувача, котрі знаходяться в різних точках міста, відносно далеко від центру, в житлових районах. Приклади кривих чистого потоку для різних типів станцій по годинно наведено на рис. 4.3.

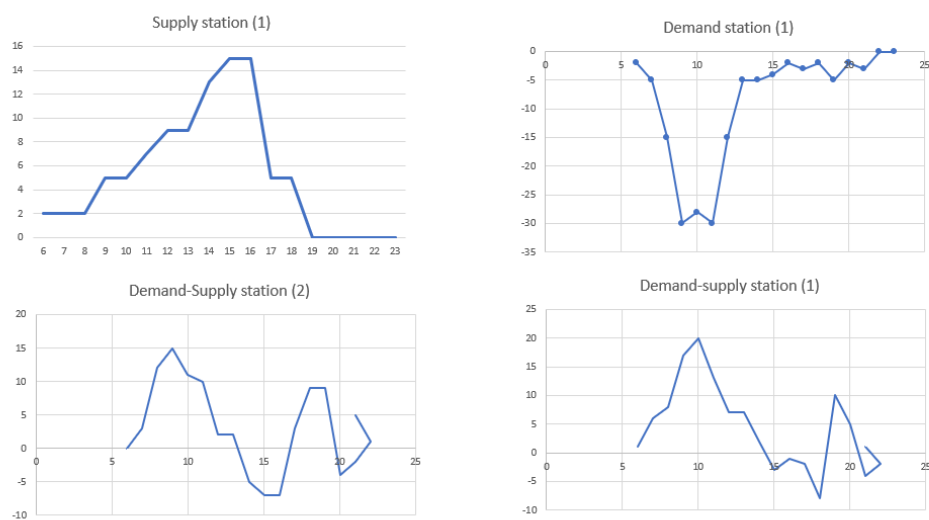


Рисунок 4.3 – Приклад кривих чистого потоку на день для станцій постачальника, отримувача та постачальника – отримувача

4.2 Прогнозування попиту на станціях

Для кожної із станцій відповідно до типу були знайдені значення $\min(H_t)$ та SD_t , де t рівне одному дню за період починаючи з 1 – го серпня до 1 – го лютого та записані в окремий файл. Приклад даних зображений на рис. 4.4.

Для кожної із станцій також було побудовано криві чистого потоку за цей період. Приклад цих кривих на рис. 4.5 – 4.7.

Day	Month	Year	Weather	Season	Peak	DayType	NetFlow	Temperature	Bikes
1	August	2017	sunny	Summer	5	Weekday	0	22	45
2	August	2017	rainy	Summer	5	Weekday	5	27	50
3	August	2017	sunny	Summer	5	Weekday	0	29	50
4	August	2017	cloudy	Summer	5	Weekday	9	22	59
5	August	2017	rainy	Summer	5	Weekend	0	25	59
6	August	2017	foggy	Summer	5	Weekend	11	23	70
7	August	2017	rainy	Summer	5	Weekday	0	19	70

Рисунок 4.4 – Приклад даних для станції постачальника

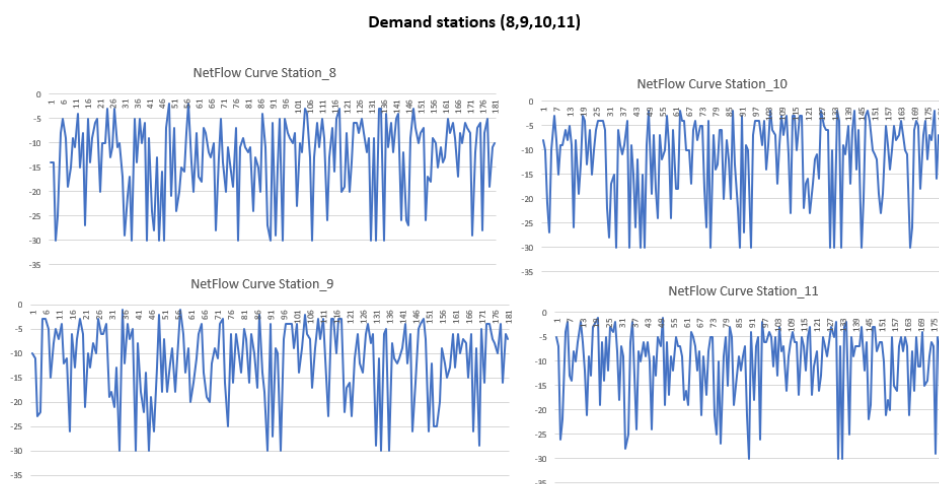


Рисунок 4.5 – Приклад кривих чистого потоку $\min(H_t)$ для терміну впродовж 6 місяців (з серпня 2017 по січень 2018 року) для станцій отримувачів

Demand - supply stations (5,6,7)



Рисунок 4.6 – Приклад кривих чистого потоку $\min(H_t)$ для терміну впродовж 6 місяців (з серпня 2017 по січень 2018 року) для станцій отримувачів – постачальників

Supply stations (1, 2, 3, 4)

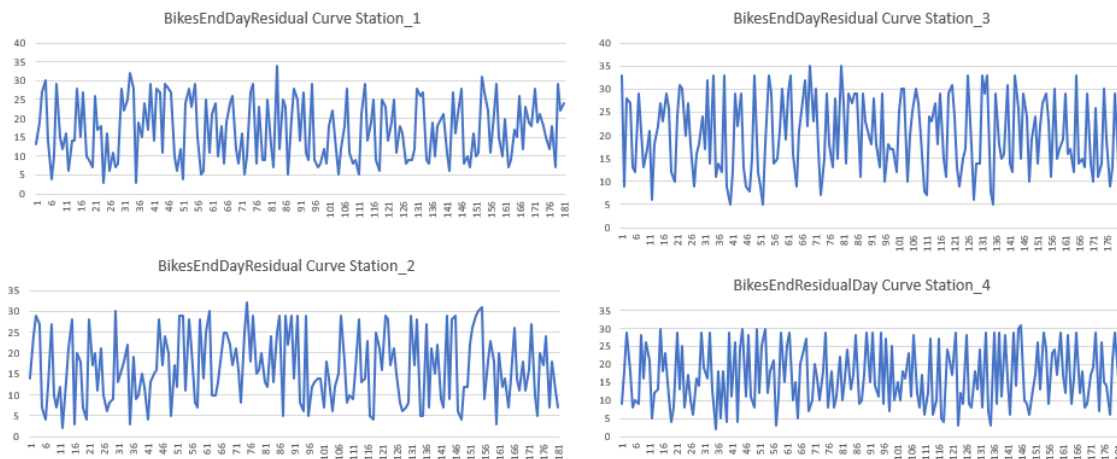


Рисунок 4.7 – Приклад кривих різниці кількості транспортних засобів вкінці та на початку дня SD_t для терміну впродовж 6 місяців (з серпня 2017 по січень 2018 року) для станцій постачальників

Для прогнозування використовувався градієнт бустинг. Був проведений експеримент з параметрами зазначеними у табл.4.1.

Таблиця 4.1 – Варіації параметрів градієнт бустингу для прогнозування

xgb1	xgb2	xgb3	param_name
0.4	0.5	1	'colsample_bytree'
1000	500	1000	'n_estimators'
3	5	3	'min_child_weight'
6	4	10	'max_depth'
0.4	0.4	0.4	'subsample'
0.01	0.05	0.1	'learning_rate'
0.02	0.02	0.02	'reg_lambda'

Для зменшення ймовірності перенавчання, а також для того аби зробити перевірку якості моделі більш точною використовувалась кросс – валідація ($k = 5$). Тобто, вибірка ділиться на 5 підвбірок, кожна із них використовується для навчання та тестування. Оцінка ефективності моделі здійснюється за середнім значенням помилки. Для порівняння також була здійснена спроба прогнозування за допомогою екстра рандомізованих дерев з параметрами зазначеними у табл .4.2.

Таблиця 4.2 – Параметри для методу Екстра рандомізованих дерев

param_name	value
n_estimators	100
min_samples_leaf	3

МАРЕ для станцій для кожної з моделей відображені у табл. 4.3 та на рис 4.8, загальне порівняння середніх значень *MAE* по усім станціям наведено на рис.4.9.

Таблиця 4.3 – MAPE з використанням певного методу для кожної із станцій

station \ MAPE	1	2	3	4	5	6	7	8	9	10	11
xgb1	0.105	0.13	0.08	0.105	0.13	0.15	0.118	0.11	0.16	0.22	0.138
xgb2	0.12	0.145	0.085	0.13	0.14	0.16	0.123	0.12	0.165	0.232	0.142
xgb3	0.113	0.135	0.083	0.128	0.135	0.15	0.122	0.115	0.167	0.225	0.14
Extra trees	0.14	0.145	0.1	0.135	0.148	0.155	0.12	0.113	0.166	0.22	0.143

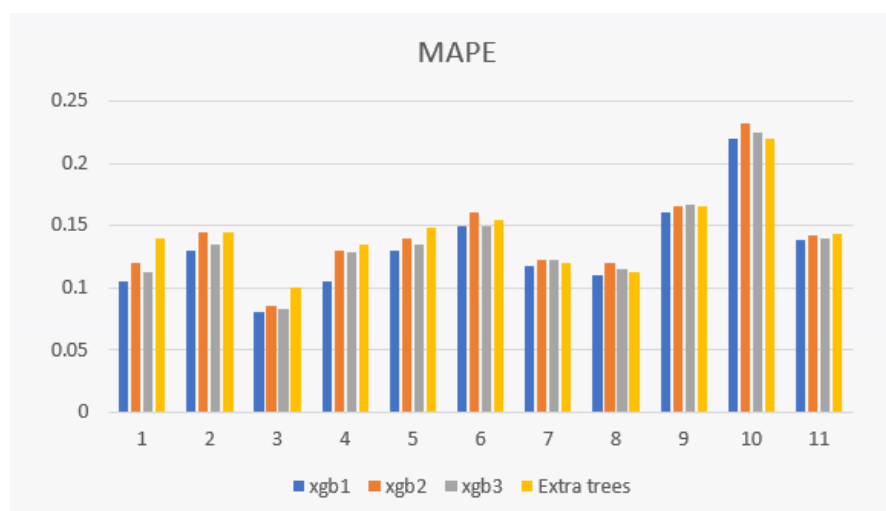


Рисунок 4.8 – MAPE для різних методів прогнозування

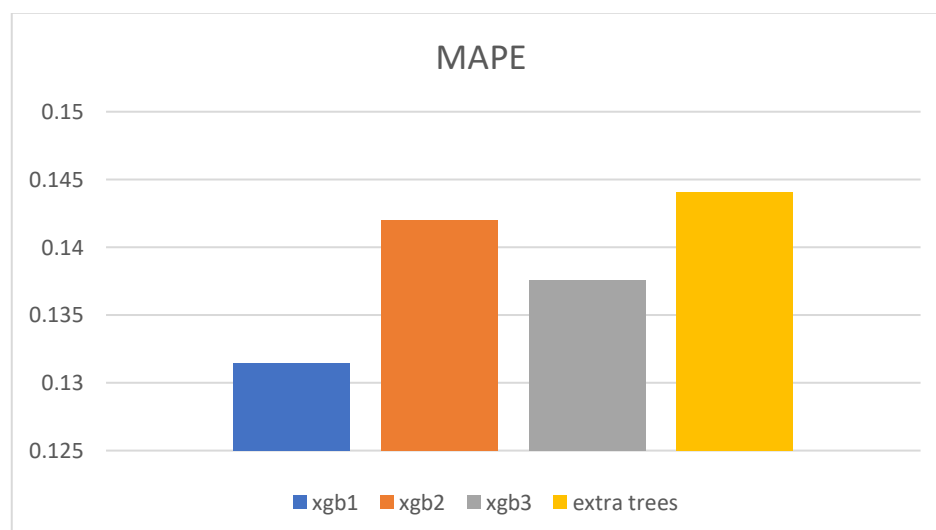


Рисунок 4.9 – Середнє значення помилки для 11 станцій з використанням різних методів прогнозування

За використання градієнт бустингу із заданими параметрами $xgb1$ результати для станцій на 1 лютого наведені у таблиці 4.4 та таблиці 4.5. Загалом у системі наявно 330 велосипедів.

Таблиця 4.4 – Результати прогнозування для станцій постачальників

Station	SD_t	N_{t-1}	q_t
1	5	40	45
2	8	35	43
3	3	39	42
4	15	41	56

Таблиця 4.5 – Результати прогнозування для станцій отримувачів та станцій постачальників – отримувачів

Station	H_t	N_{t-1}	q_t
5	- 58	56	- 2
6	- 43	60	17
7	- 44	30	- 14
8	- 18	9	- 9
9	- 19	10	- 9
10	- 15	10	- 5
11	- 13	0	- 13

Помилки становлять:

$$\text{Root Mean Squared Error} = 1.882937$$

$$\text{Mean Absolute Error} = 3.5454$$

На рис. 4.10 відображений вплив кожної із ознак на результат прогнозування для однієї із станцій.

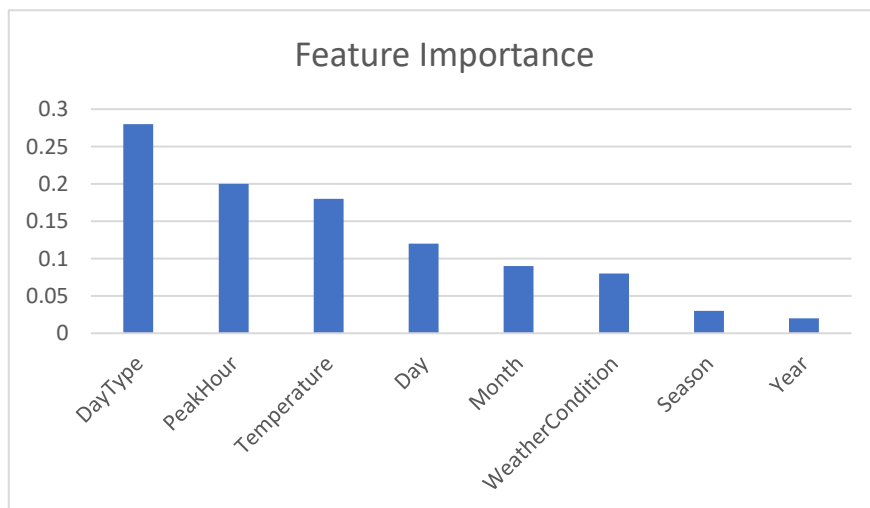


Рисунок 4.10 – Діаграма, що відображає вплив кожної із ознак на результат прогнозування

Значення середніх абсолютних процентних помилок є достатньо малими та є прийнятними для системи що має 330 транспортних засобів. Загалом даний метод прогнозування дає хороші результати для задачі статичного ребалансування.

4.3 Здійснення перепозиціонування

Система не має дефіциту, так як:

$$\sum_{i=1}^{11} q_i = 151 > 0$$

Вважатимемо, що у системі наявна певна кількість пошкоджених ТЗ, котрі необхідно забрати із станцій. Модифікуємо значення попиту для кожної станції. У випадку наявності пошкоджених ТЗ на станції постачальнику для задоволення потреб станції, достатньо аби пошкоджені велосипеди були у складі тих, що забирають. Для станції отримувача, для задоволення попиту необхідно аби на

станції була достатня кількість працюючих велосипедів, отже пошкоджені ТЗ необхідно забрати, а функціонуючі довести. Кількість пошкоджених ТЗ та модифіковані значення попиту можна побачити у табл. 4.6.

Без врахування вартості автомобілів для ребалансування маршрут для перепозиціонування зображений на рис. 4.11.

Таблиця 4.6 – Дані щодо кількості пошкоджених велосипедів та попиту на станціях

Station	1	2	3	4	5	6	7	8	9	10	11
Damaged quantity	0	13	8	0	2	10	5	1	0	3	2
Previous demand per station	45	43	42	56	-2	17	-14	-9	-9	-5	-13
Modified demand	45	30	34	56	-4	7	-19	-10	-9	-8	-15

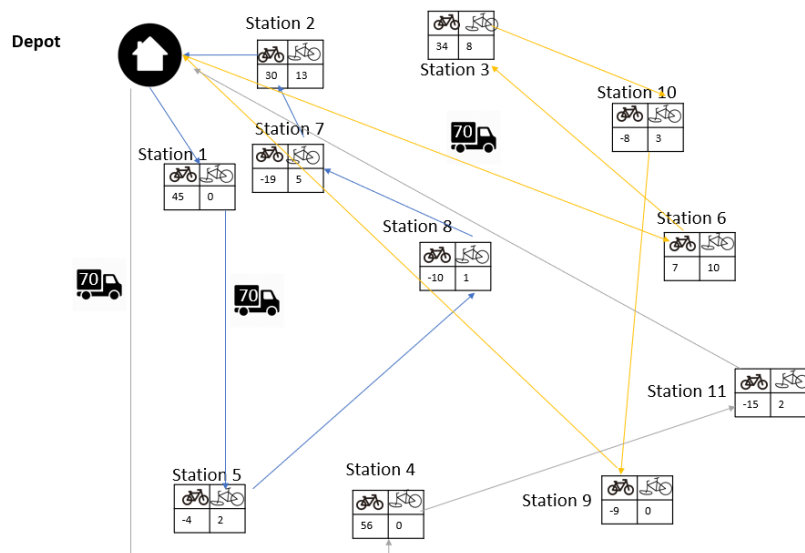


Рисунок 4.11 – Маршрут перепозиціонування для 11 станцій із врахуванням кількості пошкоджених ТЗ

При порівнянні даного підходу до репозиціонування з підходом перепозиціонування без пошкоджених ТЗ, у випадку другого, значення цільової функції буде меншим. Це пояснюється тим, що у другому випадку, автомобілям що здійснюють перепозиціонування необхідно буде забрати/довезти менше ТЗ, більш того їх місткість не буде меншати при відвідуванні будь-якої станції. Однак попит станцій з пошкодженими ТЗ фактично буде задоволений не повністю (рис. 4.12 - 4.13). Основною перевагою даного підходу у порівнянні зі звичайним перепозиціонуванням є те, що попит кожної із станцій буде задоволений на 100%. Ігнорування пошкоджених ТЗ у подальшому призведе до більшого дисбалансу на станціях, великої кількості незадоволених користувачів та накопиченню пошкоджених ТЗ на станціях.

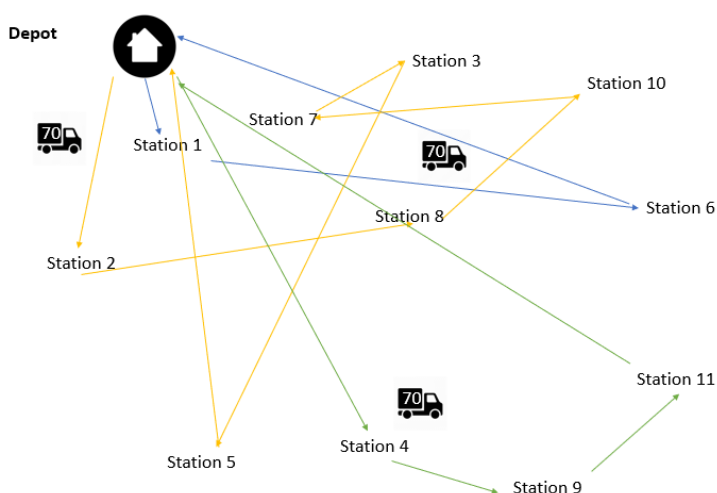


Рисунок 4.12 – Маршрут перепозиціонування для 11 станцій без врахування кількості пошкоджених ТЗ

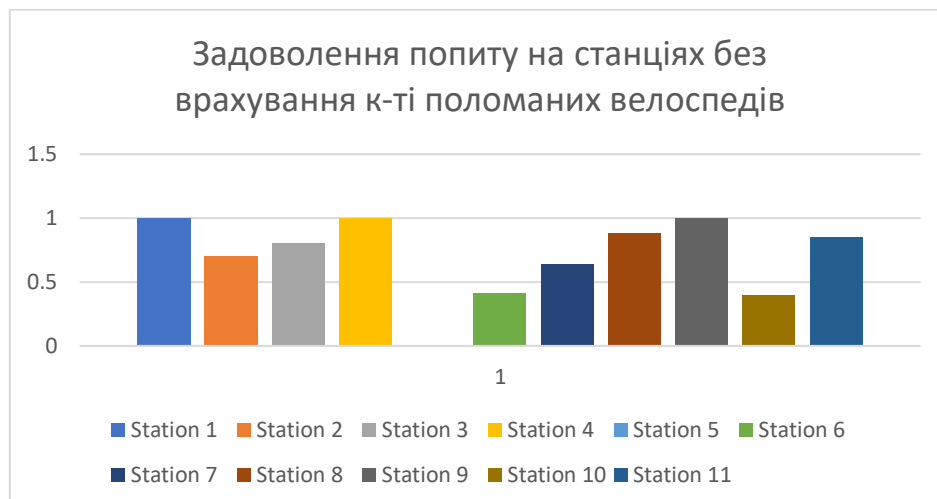


Рисунок 4.13 – Задовolenня попиту на станціях без врахування кількості пошкоджених ТЗ

Отже отримуємо 3 маршрути для перепозиціонування із врахуванням пошкоджених ТЗ:

Depot – Station₆ – Station₃ – Station₁₀ – Station₉ – Depot
Depot – Station₁ – Station₅ – Station₈ – Station₇ – Station₂ – Depot
Depot – Station₄ – Station₁₁ – Depot

Значення цільової функції наступне:

$$F_{min} = 17.4$$

Кількість задіяних автомобілів – 3.

Значення цільової функції у випадку звичайного перепозиціонування без врахування кількості пошкоджених ТЗ:

$$F_{min} = 15.5$$

Кількість задіяних автомобілів – 3.

Розглянемо цю ж задачу перепозиціонування із врахуванням особливостей парку автомобілів.

Значення ΔL та ΔT для усіх вантажних автомобілів – вантажних бортових автомобілів, вантажних автомобілів – фургонів, автомобілів – самоскидів, автомобілів – тягачів є однаковим:

$$\Delta T = 0.077$$

$$\Delta L = 0.0023$$

Дані про наявність автомобілів у автопарку, їх характеристики та коефіцієнти наведені у табл. 4.7 – табл. 4.8.

Таблиця 4.7 – Дані щодо характеристик автомобілів, що здійснюють ребалансування

Авто	Кількості років експлуатації	Пробіг (км)	Вартість на ринку у грн.	Середня вартість обслуговування на рік
Mercedes – Benz (2018)	2	356000	587100	7200
Mercedes – Benz (2014)	6	330000	456330	7200
Renault Master	5	356000	396150	9000
Opel Movano (2012)	8	260000	344850	11000
Opel Movano (2010)	10	396000	298330	11000

Таблиця 4.8 – Дані щодо цінових коефіцієнтів кожного із автомобілів

Авто	$Price_1$	$Price_2$	$Price_3$	Sum
Mercedes – Benz Sprinter (2018)	1.616	0.040	3.5	5.156
Mercedes – Benz Sprinter (2014)	1.355	0.1309	2.8	4.2859
Renault Master	1.112	0.126	2.9	4.138
Opel Movano (2012)	1.326	0.338	3.5	5.164
Opel Movano (2010)	0.75	0.277	3.7	4.727

За результатами найбільш вдалим для ребалансування виявився автомобіль *Renault Master*. Маршрут виявився таким же, як і з врахуванням кількості пошкоджених ТЗ, що пояснюється приблизно однаковими ціновими коефіцієнтами авто. Однак для здійснення перепозиціонування були вибрані певні авто, а саме *Mercedes – Benz Sprinter (2014)* для другого маршруту, *Renault Master* для першого маршруту та *Opel Movano (2010)* для третього маршруту. Це автомобілі з найменшою ціною використання. Для порівняння, у випадку використання випадкових автомобілів, для здійснення перепозиціонування. Нехай, для прикладу, авто *Opel Movano (2012)* для першого маршруту, авто *Mercedes – Benz Sprinter (2018)* для другого маршруту та *Opel Movano (2010)* для третього маршруту ціна ребалансування була б відносно більшою на 10%. Тому врахування характеристик автопарку є дійсно важливим аби мінімізувати затрати на перепозиціонування. Більш того, врахування автомобілів необхідних для ребалансування дасть можливість більш чітко спланувати подорож та попередити можливі проблеми (відсутність авто у автопарку, поломки авто).

Таким чином даний підхід, у порівнянні із традиційними, основною ціллю яких є мінімізація лише пройденого шляху дає можливість уникнути зайвого перевантаження станцій та затрат на обслуговування конкретних автомобілів, а також позбавляє від накопичення в ССВМТЗ пошкоджених ТЗ, що тягнуть за собою невірне прогнозування попиту станцій, котре призведе до нестачі ТЗ на станціях та більшої кількості незадоволених запитів користувачів.

Оцінку ефективності ребалансування проведемо за критерієм (2.22). Виділимо окремі тестові вибірки для Грудня 2017 р. та Січня 2018 р. Початкове прогнозування відповідно здійснюватимемо використовуючи дані Серпень – Листопад для першої вибірки та Серпень – Грудень для другої вибірки. Загальним підходом до ребалансування у маленьких/середніх містах вважається розподіл порівну ТЗ вкінці дня. За обрахунком кількості незадоволених запитів метод ребалансування представлений у даній роботі показав набагато кращі результати, аніж загальний підхід до ребалансування згаданий вище. Результати наведені на рис. 4.14 - 4.15.

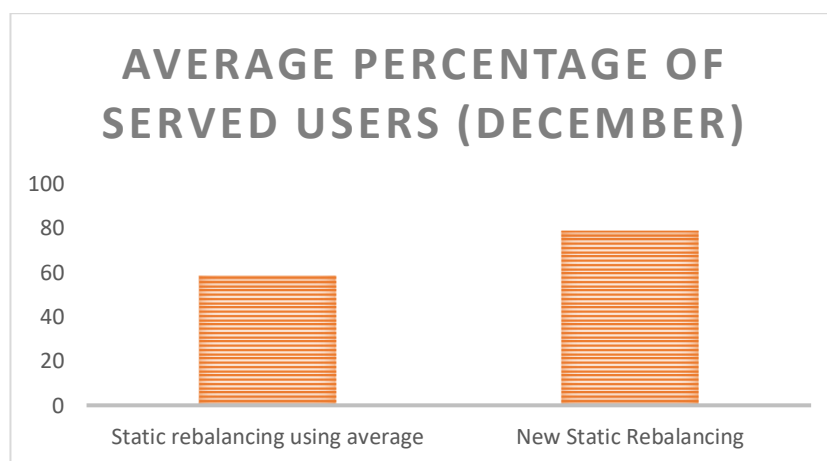


Рисунок 4.14 – Середній відсоток користувачів, запити котрих були задоволені (Грудень 2017)

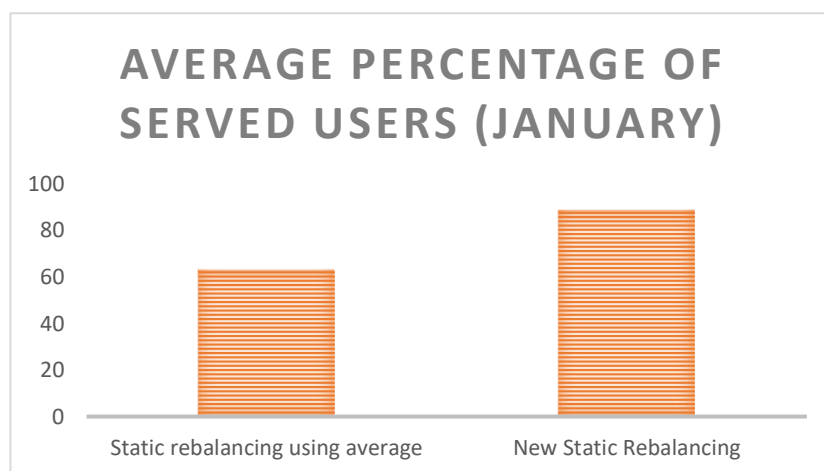


Рисунок 4.15 – Середній відсоток користувачів, запити котрих були задоволені (Січень 2018)

4.4 Висновки

У даному розділі було детально розглянуто роботу програмного продукту. Було надано інформацію про формат вхідних даних, представлено результати розподілу станцій на отримувачі, постачальники, а також постачальники – отримувачі та представлені криві чистого потоку для кожної із станцій. Було детально описано процедуру прогнозування. Для початку було розглянуто використання декількох методів, котрі можуть бути використанні для прогнозування попиту на станціях, а саме градієнтного бустингу з різною варіацією параметрів та екстра рандомізованих дерев, наведено похибки для кожного із методів. Здійснено порівняння результатів роботи цих методів. Найкращим виявився градієнт бустинг, тому для прогнозування використовувався саме він. Були представлені результати прогнозування, знаходження попиту та модифікації попиту відповідно до кількості пошкоджених ТЗ для кожної із станцій. Особлива увага приділялась розгляду задачі перепозиціонування. Для початку була зроблена перевірка чи наяна достатня кількість ТЗ у системі. Потім було розглянуто задачу

перепозиціонування з та без врахування кількості пошкоджених ТЗ, представлені маршрути, значення цільових функцій, здійснене порівняння двох підходів до перепозиціонування. Пізніше було розглянуто задачу перепозиціонування з врахуванням характеристик автомобілів, що власне здійснюють цю операцію, наведені результати порівняння двох підходів та основні переваги використання даного підходу.

Перепозціонування з усуненням нефункціонуючих ТЗ виявилось більш затратним, однак врахування таких ТЗ відіграє надзвичайно важливу роль. Наявність пошкоджених ТЗ на станціях призведе до неповного задоволення попиту, а отже втрат коштів та користувачів. Ігнорування наявності таких ТЗ призведе до їх накопичення на станціях, що значно вплине на задоволення попиту, а пізніше і на репутацію байкшерінг компанії.

Також, врахування характеристик автомобілів дасть змогу зробити ребалансування менш затратним. Частина бюджету байкшерінг компаній витрачається безпосередньо на обслуговування авто. Правильний підбір авто дасть змогу мінімізувати ці витрати.

Також була проведена оцінка ефективності даного методу ребалансування, котра визначалась середнім відсотком задоволених запитів користувачів. Даний метод виявився значно ефективнішим, аніж загальний підхід щодо розподілу ТЗ порівну між станціями, що застосовується для маленьких міст.

РОЗДІЛ 5 РОЗРОБЛЕННЯ СТАРТАП – ПРОЕКТУ

В останні роки набув великої популярності такий вид малого підприємництва як стартап. Стартап проект – є комерційним проектом, який знаходиться в стані розробки, або нещодавно вийшов на ринок. Характерною особливістю стартапу, що відрізняє його від малого бізнесу, є оригінальність та інновації, він не може бути копією вже реалізованих ідей. При цьому проект не обов'язково повинен бути масштабного характеру, головне, щоб він був креативним, а його завдання – спрощувати людям будьякі дії в їх повсякденному житті.

Наразі, з появою Інтернету та сучасних технологій, стало простіше заходити на ринок, знаходити інвесторів та споживачів. З'явилося набагато більше можливостей для розвитку свого проекту за кордоном, ніж раніше. Проте розробка стартапу є досить ризикованим завданням. Не всім вдається довести свій стартап проект до ринкового впровадження. За статистикою успіху досягає лише 10 – 20% від усіх стартап проектів. Запуск стартапу передбачає цілий ряд обов'язкових дій, в межах яких визначають ринкові перспективи стартапу, графік розробки, принципи організації виробництва, заходи з залучення інвесторів та аналіз ризиків.

5.1 Опис ідеї стартап проекту

Назва стартап проекту: “Система ефективного ребалансування мобільних транспортних засобів”.

У табл. 5.1 подано зміст ідеї стартап проекту, можливі напрямки застосування та основні вигоди, що може отримати користувач товару. У табл. 5.2 визначені сильні, слабкі та нейтральні сторони проекту.

Таблиця 5.1 – Опис ідеї стартап – проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Система ефективного ребалансування мобільних транспортних засобів, котра за вхідними даними зі станцій, а також за даними кількості ресурсів пропонує схему для ребалансування та вибирає найоптимальніший маршрут	Використання ССВМТЗ	Система дозволяє швидко визначити оптимальну кількість велосипедів для кожної станції, а також оптимальний маршрут перепозиціонування

Таблиця 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ п/п	Технічно – економічні характеристики ідеї	(Потенційні) товари/концепції конкурентів		
		Мій проект	Logvpr	Open – VPR
1.	Ціна	Низька	Висока	Низька
2.	Функціонал	Широкий	Широкий	Вузький

Отже, з табл. 5.2 можна визначити, що ціна та функціонал є сильними характеристиками для потенційного товару.

5.2 Технологічний аудит ідеї проекту

За результатами аналізу табл. 5.3 можна зробити висновок про можливість технологічної реалізації проекту.

Таблиця 5.3 – Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1.	Система ефективного ребалансування мобільних транспортних засобів	Динамічне ребалансування, прогнозування на основі нейронних мереж (Python, TensorFlow, MongoDB, C++, AngularJS)	Наявна	Недоступна
2.	Система ефективного ребалансування мобільних транспортних засобів	Статичне ребалансування, прогнозування на основі градієнтного бустингу (Python, C++, XGBoost, CPLEX, SQLServer)	Наявна	Доступна
Обрана технологія реалізації ідеї проекту: Статичне ребалансування, прогнозування на основі градієнтного бустингу(Python, C++, XGBoost, CPLEX, SQLServer)				

5.3 Аналіз ринкових можливостей запуску стартап проекту

5.3.1 Аналіз попиту та потенційних груп клієнтів

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту та ринкових загроз, які можуть перешкодити реалізації проекту дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів конкурентів.

Попередню характеристику ринку можна побачити у табл.5.4.

Таблиця 5.4 – Попередня характеристика потенційного ринку стартап – проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1.	Кількість головних гравців, од	15
2.	Загальний обсяг продаж, грн/ум.од	100 000 ум.од.
3.	Динаміка ринку (якісна оцінка)	Зростає
4.	Наявність обмежень для входу (вказати характер обмежень)	Немає обмежень для входу
5.	Специфічні вимоги до стандартизації та сертифікації	Немає специфічних вимог до стандартизації та сертифікації
6.	Середня норма рентабельності в галузі (або по ринку), %	75 %

Середня норма рентабельності галузі вища за банківський відсоток на вкладення, що означає привабливість ринку для входження за попереднім оцінюванням. Визначимо потенційні групи клієнтів, їх характеристики, та сформуємо орієнтовний перелік вимог до товару для кожної групи (табл. 5.5).

Таблиця 5.5 – Характеристика потенційних клієнтів стартап – проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1.	Потреба спрогнозувати попит станцій наступного дня	Компанії, котрі уже мають певний підхід до ребалансування, однак потребують більшої точності прогнозу	Клієнту надається лише частина продукту, а саме модуль прогнозування	Простота у використанні. Висока точність прогнозування

Продовження таблиці 5.5

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
2.	Потреба у використанні кожного дня, а саме щоденного прогнозування та знаходження оптимального маршруту для ребалансування	Нові компанії, котрі не мають своєї системи ребалансування, або ж компанії системи котрих не є ефективними	Клієнт використовує усі можливості системи, як результат отримує кінцевий оптимальний маршрут	Зручний інтерфейс. Ефективний підхід до ребалансування з врахуванням велосипедів у аварійному стані та характеристик автопарку. Швидкодія (великі об'єми даних)

5.3.2 Аналіз ринкового середовища

Проведемо аналіз ринкового середовища. Нижче наявні таблиці факторів, що сприяють ринковому впровадженню проекту та факторів, що йому перешкоджають (табл. 5.6 – 5.7).

Таблиця 5.6 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1.	Конкуренція	Реліз нової версії системи Logvpr з вдосконаленою аналітикою для користувачів	Вдосконалити або модифікувати алгоритм та аналітику платформи

Продовження таблиці 5.6

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
2.	Постачальники	Постачальники планують підвищити ціну на ліцензійне програмне забезпечення, необхідне для реалізації та підтримки проекту	Залучити інвестиції з урахуванням змін, вести перемовини з постачальниками з приводу фіксування ціни за умов довготривалої співпраці
3.	Збут	Лояльність споживачів до існуючих конкурентів може призвести до низької зацікавленості у власному продукті	Потрібно розробити вдалі маркетингову та рекламну кампанії
4.	Зміна потреб користувача	Користувачам необхідні рішення з іншим функціоналом	Передбачення можливості додавання нового функціоналу (Наприклад такого, як врахування екофактору, а саме викиду шкідливих речовин у атмосферу)

Таблиця 5.7 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1.	Конкуренція	Збільшення цін на тарифи користування системою Logvtr	Демпінгувати ціни на послуги. Побудова рекламних кампаній
2.	Постачальники	Поява нових постачальників, з більш вигідними пропозиціями ресурсів	Налагодити комунікацію з потенційними партнерами
3.	Збут	Великі компанії конкуренти планують підвищити ціну за свої послуги	Розглядати стратегію демпінгування ціни
4.	Поява нових цільових груп клієнтів	Потреба в аналогічному продукті в інших сферах діяльності	Адаптація продукту під нові сфери використання

5.3.3 Аналіз пропозиції

Проведемо аналіз пропозиції: визначимо загальні риси конкуренції на ринку (табл. 5.8).

Таблиця 5.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Тип конкуренції – монополістична конкуренція	Ринок складається з відносно великої к-ті продавців, кожен з яких володіє невеликою (не нескінченно малою) часткою ринку; встановлюючи ціни, продавці намагаються виділитися за неціновими ознаками; товар продавця є недосконалим замінником товарів інших фірм; ринок не має бар'єрів входу/виходу	Для того, щоб відрізнитися від конкурентів чимось, окрім ціни, продавці розробляють різні пропозиції для різних сегментів споживачів і широко використовують практику використання марок, рекламу та методи особистих продажів. Також необхідною є підтримка якості продукту та постійні вдосконалення
За рівнем конкурентної боротьби – інтернаціональний рівень	Конкуренти діють на інтернаціональному просторі – в мережі Інтернет	Забезпечити можливість користування послугами (продуктом) незалежно від місцезнаходження. Підтримувати продукт на національному ринку
За галузевою ознакою – внутрішньогалузева	Має місце суперництво між окремими підприємцями і фірмами однієї галузі щодо одержання прибутку	Підвищення продуктивності праці, зменшення витрат виробництва, зниження індивідуальної цінності товару, розширення функціоналу продукту, вдосконалення для застосування у нових галузях (для нових видів перевозок, транспорту)

Продовження таблиці 5.8

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Конкуренція за видами товарів – товарно – видова	Це конкуренція між товарами одного виду – різні рекомендаційні системи виконують один і той же набір функцій	Використовувати цінові та нецінові методи конкурентної боротьби на ринку
За характером конкурентних переваг – нецінова	Проводиться головним чином за допомогою вдосконалення якості продукції, технології виробництва, інновацій та нанотехнологій, патентування, брендування, умов її продажу.	Випускати новий товар (послугу), які принципово відрізняються від своїх попередників або представляють модернізований варіант старої моделі
За інтенсивністю – не марочна	Роль торгової марки незначна, хоча самі марки можуть бути присутніми на ринку	Немає необхідності вкладати кошти у створення та розкрутку бренду, необхідно приділяти увагу якості продукту а не бренду компанії

Після аналізу конкуренції проведемо більш детальний аналіз умов конкуренції в галузі (за моделлю 5 сил М. Портера) (табл. 5.9).

Таблиця 5.9 – Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари – замітники
Складові аналізу	Logvtr, Open VRP, Вусycle, GoVike	NextBike, Reflex	Ціна на ресурси; Готовність до співпраці; Якість ресурсів	Вплив покупців на ринкову ціну; Очікування щодо якості обслуговування	Поява відносно дешевих або безкоштовних заміників
Висновки	З боку прямих конкурентів очікується інтенсивна конкурентна боротьба. GoVike відкритий до співпраці	– є можливість виходу на ринок, бар'єри відсутні; – потенційні конкуренти – великі компанії з потужними виробничими можливостями; – загалом названі вище компанії мають декілька вже успішних реалізованих варантів схожої продукції та готують інновації, тому потенційний ріст популярності продукції компаній невідомий	Постачальники контролюють ціни на ресурси та їхню якість	Клієнти диктують умови роботи на ринку	Є схожі товари, однак дана версія є унікальною в певному роді, саме таких безкоштовних аналогів наразі не існує

Отже, з огляду на конкурентну ситуацію у галузі, бачимо, що є можливість входження на ринок. Щоб бути конкурентноспроможним на ринку, проект має бути привабливим за ціною, якістю роботи та мати зручний інтерфейс. Зокрема проект дещо відрізняється від вже існуючих систем, а саме враховує можливі поломки транспорту на станціях та ціну використання конкретного автомобіля для ребалансування.

5.3.4 Фактори конкурентоспроможності

У табл. 5.10 наведено фактори конкурентноспроможності, сформовані на основі аналізу конкуренції та з урахуванням характеристик ідеї проекту, вимог споживачів до товару (табл. 5.5) та факторів маркетингового середовища (табл. 5.6 – 5.7)

Таблиця 5.10 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1.	Технологія	Рівень застосовуваних технологій відіграє значну роль в побудові ефективної системи ребалансування, а отже надання якісного сервісу споживачу
2.	Ціна	Ціна має значення як для приватних особи – користувача онлайн – системи, так і для великих компаній, які зацікавлені у придбанні абонементу на користування ефективною системою ребалансування

Продовження таблиці 5.10

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
3.	Маркетинг	<ul style="list-style-type: none"> • Частка ринку, займаного підприємством; • Престиж торгової марки (бренд); • Витрати на стимулювання збуту і їх ефективність
4.	Репутація	<ul style="list-style-type: none"> • Репутація (імідж) підприємства; • Якість обслуговування
5.	Унікальність позиціонування	В умовах монополістичної конкуренції, коли фактор диференціації торгової марки є ключовим засобом ведення конкурентної боротьби, важливим є створення та підтримання унікального позиціонування, що створює певний захист від конкурентних зіткнень. Для поточного продукту наявна інновація, котра відрізняє його від вже існуючих продуктів на ринку

5.3.5 Аналіз сильних та слабких сторін стартап – проекту

У табл. 5.11 наведено аналіз сильних та слабких сторін проекту, сформульованих за визначеними факторами конкурентоспроможності (табл. 3.10).

Серед основних компаній конкурентів можна виділити такі як:

1. Logvtr
2. OpenVRP
3. Vucycle
4. GoBike

Таблиця 5.11 – Порівняльний аналіз сильних та слабких сторін «Система побудови рекламних кампаній»

№ п/п	Фактор конкурентоспроможності	Бали 1 – 20	Рейтинг товарів – конкурентів у порівнянні із товаром стартапу						
			-3	-2	-1	0	1	2	3
1.	Технологія	20					1	2	3/4
2.	Ціна	10						1/2	3/4
3.	Маркетинг	9	1/2	3			4		
4.	Репутація	8		1		2/3	4		
5.	Унікальність	19					1	2/3	4

5.3.6 SWOT – аналіз

У табл 5.12 наведено SWOT – аналіз (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (табл. 5.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (табл. 5.11).

Таблиця 5.12 – SWOT – аналіз стартап – проекту

<p>Сильні сторони:</p> <ol style="list-style-type: none"> 1. Ціна нижча за ціну конкурентів. 2. Унікальне позиціонування – як в якості відокремленого ПЗ для прогнозування так і в якості системи для ребалансування. 3. Наявна інновація, що впливає на ефективність системи. 	<p>Слабкі сторони:</p> <ol style="list-style-type: none"> 1. Технологія відстає від ринкових лідерів. 2. Відсутність чітко вираженої маркетингової стратегії, непослідовність в її реалізації. 3. Нейтральна репутація виробника.
<p>Можливості:</p> <ol style="list-style-type: none"> 1. Можливість зміцнення іміджу підприємства за допомогою реклами та маркетингу. 2. Можливість перетворення головних конкурентів на партнерів (споживачів) – шляхом постачання абонементу на користування системою та надання ефективних рекомендацій. 	<p>Загрози:</p> <ol style="list-style-type: none"> 1. Загроза працювати без прибутку внаслідок демпінгування цін. 2. Загроза втрати споживачів внаслідок підвищення конкуренції з боку великих компаній – конкурентів в галузі. 3. Загроза підвищення цін на користування послугами внаслідок підвищення цін на ресурси – як результат втрата покупців залучених низькими цінами.

На основі SWOT аналізу визначимо альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок (табл. 5.13).

Таблиця 5.13 – Альтернативи ринкового впровадження стартап – проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1.	Впровадження існуючого товару на існуючому ринку	Середня	4 міс
2.	Розвиток товару на існуючому ринку	Висока	6 міс
3.	Розвиток нового ринку з існуючим товаром	Низька	5 міс
4.	Впровадження нового товару і створення нового ринку	Дуже низька	8 міс

Обрано альтернативу 2, для якої ймовірність отримання ресурсів є високою і строк реалізації відносно невеликий.

5.4 Розроблення ринкової стратегії проекту

5.4.1 Вибір цільових груп потенційних споживачів

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (табл. 5.14).

Таблиця 5.14 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1.	Компанії, котрі уже мають певний підхід до ребалансування, однак потребують більшої точності прогнозу, так як попереднє ребалансування не є ефективним.	Низька	Низький	Висока	Середня
2.	Нові компанії, котрі не мають своєї системи ребалансування або ж компанії чиї системи ребалансування є неефективними.	Висока	Високий	Середня	Висока
Які цільові групи обрано: Нові компанії, котрі не мають своєї системи ребалансування.					

5.4.2 Базова стратегія розвитку

У табл. 5.15 наведено базову стратегію розвитку.

Таблиця 5.15 – Визначення базової стратегії розвитку

№ п/ п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
	Розвиток товару на існуючому ринку	Стратегія концентрованого маркетингу	Позиціонування за співвідношенням "ціна – якість", позиціонування на основі порівняння товару фірми з товарами конкурентів.	Стратегія спеціалізації

5.4.3 Стратегія конкурентної поведінки

У табл. 5.16 наведено стратегію конкурентної поведінки.

Таблиця 5.16 – Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, які?	Стратегія конкурентної поведінки
1.	Ні.	Забирати існуючих у конкурентів	Так, з додаванням унікальних властивостей, суттєвим покращенням ефективності, характеристик продукту	Стратегія наслідування лідера

5.4.4 Стратегія позиціонування

На основі вимог споживачів з обраних сегментів до постачальника (стартап – компанії) та до продукту (див. табл. 5.5), а також в залежності від обраної базової стратегії розвитку (табл. 5.15) та стратегії конкурентної поведінки (табл. 5.16) розробляється стратегія позиціонування (табл. 5.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект.

Таблиця 5.17 – Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап – проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1.	Зручність у використанні (наявність простого та зручного користувацького інтерфейсу), ефективність, швидкість роботи, ціна .	Стратегія спеціалізації	Ефективне ребалансування засноване на вдалому передбаченні попиту та врахуванні кількості пошкоджених велосипедів.	1. Позиціонування на основі порівняння товару фірми з товарами конкурентів; 2. Позиціонування на позитивних особливостях технології; 3. Позиціонування за показниками якості.

5.5 Розроблення маркетингової програми стартап – проекту

5.5.1 Ключові переваги концепції потенціоного товару

У табл. 5.18 наведено результати попереднього аналізу конкурентоспроможності товару.

Таблиця 5.18 – Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1.	Потреба системи у ребалансуванні	Результуючий маршрут є оптимальним. У системі враховано можливість наявності транспортних засобів у аварійному стані. Тобто система ребалансування передбачає виявлення та заміну таких засобів без необхідності у зайвих подорожах.	Більш точні прогнози, оптимальний маршрут без необхідності додаткових затрат на вирішення задачі заміни аварійних транспортних засобів, зручний інтерфейс, низька ціна, мінімізація коштів за рахунок врахування характеристик автопарку.

5.5.2 Трирівнева маркетингова модель товару

Надалі розробляється трирівнева маркетингова модель товару: уточнюється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання (табл. 5. 19).

Таблиця 5.19 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Система ефективного ребалансування мобільних транспортних засобів, котра за вхідними даними зі станцій, а також за даними кількості ресурсів пропонує схему для ребалансування та вибирає найоптимальніший маршрут. Врахування наявності аварійних транспортних засобів, швидкодія, зручний інтерфейс.		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх/Тл/Е/Ор
	1. Система ефективного ребалансування мобільних транспортних засобів, котра за вхідними даними зі станцій, а також за даними кількості ресурсів пропонує схему для ребалансування та вибирає найоптимальніший маршрут із врахуванням наявності аварійних транспортних засобів.	N/A	N/A
	Якість: тестування, дослідження ефективності прогнозування з використанням декількох метрик (показників) якості, дослідження ефективності ребалансування.		
	Пакування: N/A		
	Марка: "SmartRebalancing"		
III. Товар із підкріпленням	До продажу: дослідження ефективності, тестування.		
	Після продажу: онлайн – підтримка, гарантійне обслуговування, навчання персоналу.		
За рахунок чого потенційний товар буде захищено від копіювання: Ліцензія. Вихідний код програмного продукту є закритим, та не передається клієнтам і третім особам. На програмний продукт оформлено авторське право.			

5.5.3 Визначення цінових меж

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово – економічного аналізу проекту), яке передбачає аналіз ціни на товари – аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (табл. 5.20). Аналіз проводиться експертним методом.

Таблиця 5.20 – Визначення меж встановлення ціни

№ п/п	Рівень цін на товари – замітники	Рівень цін на товари – аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
	2000\$ / рік	1000 – 1500\$ / рік	90000\$ / рік	Базова покупка та впровадження: нижня межа 1000\$, верхня межа 2000\$.

5.5.4 Формування системи збуту

У таблиці 5.21 наведено формування системи збуту.

Таблиця 5.21 – Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1.	Цільові клієнти – ССВМТЗ, державні установи на місцях, які бажають впровадити у своїй роботі сучасні ефективні засоби. Вони цікавляться інноваційними рішеннями, відвідують тематичні семінари та конференції. Цільові клієнти купують ключ для ПЗ, який є дійсним 1 рік.	Формування попиту і стимулювання збуту. Встановлення контактів із споживачами. Просування маркетингової інформації. Забезпечення безпечної грошової транзакції.	Нульова або однорівнева (сервіс безпосередньо продається споживачам та через посередників)	Власні засоби збуту

5.5.5 Концепція маркетингових комунікацій

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позицінування, визначену специфіку поведінки клієнтів (табл. 5.22).

Таблиця 5.22 – Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1.	Цільові клієнти з великою ймовірністю вже користуються певними подібними сервісами.	Рекомендації знайомих, прямий зв'язок з цільовими клієнтами.	Позиціонування на основі порівняння товару фірми з товарами конкурентів; Позиціонування на позитивних особливостях технології; Позиціонування за показниками якості.	Привернути увагу споживача, ознайомити з усіма перевагами використання продукту, переконати в унікальності продукту.	Презентація переваг та співвідношення ціна – якість. Зменшуємо затрати на ребалансування, враховуємо всі необхідні умови та мінімізуємо кількість незадоволених клієнтів.

Отже було сформовано ринкову (маркетингову) програму, що включає в себе концепції товару, збуту, просування та попередній аналіз можливостей ціноутворення, спирається на цінності та потреби потенційних клієнтів, конкурентні переваги ідеї, стан та динаміку ринкового середовища, в межах якого буде впроваджено проект, та відповідну обрану альтернативу ринкової поведінки.

5.6 Висновки

В даному розділі проведено аналіз створення та виведення на ринок стартаппроекту на основі програмного продукту, який було розроблено в рамках магістерської дисертації. В межах цього аналізу було розроблено опис самої ідеї проекту, визначено загальні напрями використання товару, проаналізовано ринкові можливості щодо впровадження проекту, визначено відмінності від конкурентів та розроблено стратегію виходу на ринок. Узагальнюючи проведений аналіз, можна зазначити, що є можливість ринкової комерціалізації проекту. Наявний попит, динаміка ринку зростає, а також висока рентабельність роботи на ринку. З огляду на потенційні групи клієнтів та високий рівень конкурентоспроможності проекту є достатні перспективи для впровадження стартапу. Отже, подальша імплементація проекту є доцільною.

ВИСНОВКИ

У даній магістерській дисертації було розглянуто проблему ребалансування ССВМТЗ. Для розв'язку поставленої проблеми були проаналізовані основні методи, що застосовуються для ребалансування систем, розглянуті їх переваги та недоліки. Особлива увага приділялася статичному ребалансуванню. Були розглянуті можливі варіанти прогнозування попиту на станціях та здійснення перепозиціонування ТЗ.

Детально був розглянутий градієнт бустинг для задачі регресії. Саме цей метод застосовувався для прогнозування попиту на станціях у даній роботі. Також розглядалось вирішення задачі репозиціонування за допомогою методу гібридизації дискретного рою часток та локального пошуку зі змінними околами. Було детально розглянуто вибраний підхід до здійснення прогнозування на станціях.

Був запропонований новий метод для здійснення статичного ребалансування котрий відрізняється від наявних тим, що при проведенні операції перепозиціонування враховує наявність пошкоджених транспортних засобів у системі, а також ціну використання конкретного автомобіля. Перепозиціонування з усуненням нефункціонуючих ТЗ відіграє надзвичайно важливу роль, так як наявність пошкоджених ТЗ на станціях призведе до неповного задоволення попиту, а отже втрат коштів та користувачів. Щодо врахування характеристик автомобілів, то такий підхід дасть змогу зробити ребалансування менш затратним.

Також була проведена оцінка ефективності даного методу ребалансування, котра визначалась середнім відсотком задоволених запитів користувачів. Даний метод виявився значно ефективнішим, аніж загальний підхід щодо розподілу ТЗ порівну між станціями, що застосовується для маленьких міст.

В якості практичного прикладу застосування, було розроблено програмний продукт з використанням технологій Python та C++. Було проведено аналіз

роботи програмного продукту із застосуванням розробленого методу, наведені результати прогнозування попиту та маршрути для перепозиціонування.

В подальших дослідженнях рекомендується дослідити можливість врахування екологічного фактору в моделі задля зменшення негативного впливу ребалансування на навколишнє середовище, а також можливість здійснення ребалансування за допомогою автомобілів із різною місткістю. Важливою є також більш точна оцінка вартості використання конкретного автомобіля, що використовується для перепозиціонування за рахунок додаткового врахування його певних характеристик. Також рекомендується використати можливість врахування пошкоджених велосипедів та характеристик автомобілів, що здійснюють перепозиціонування для задачі динамічного ребалансування. Також необхідно врахувати можливість суміщення динамічного та статичного ребалансування зі здійсненням гейміфікації системи, а саме створення додатку для взаємодії з користувачем напряду та його заохочення до підтримки системи у необхідному стані.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Dantzig A.G.B., Ramser J.H., Science S.M., and Oct N. The Truck Dispatching Problem Stable. *Management*. 2010. Vol. 6, No. 1. P. 80 – 91.
2. P. DeMaio. Bike – sharing: History, Impacts, Models of Provision, and Future. *Journal of Public Transportation*. 2009. Vol. 4, No.12. P.41 – 56.
3. OBIS (Optimizing bike sharing in European cities). *Project Handbook*. URL: <https://ec.europa.eu/energy/intelligent/projects/en/projects/obis> (last access date: 20.11.2020).
4. R. Meddin. The Meddin Bike – sharing World Map. *World Maps*. URL: <https://bikesharingworld.com> (last access date: 18.09.2020).
5. LDA Consulting, Capital Bikeshare Member Survey Report 2011. URL: [https://www.capitalbikeshare.com/system – data](https://www.capitalbikeshare.com/system-data) (last access date: 18.11.2020).
6. Kaltenbrunner A, Meza R, Grivolla J, Codina J, Banchs R. Urban cycles and mobility patterns: exploring and predicting trends in a bicycle – based public transport system. *Pervasive and Mobile Computing*. 2010. Vol.6, No.4. P.455 – 466.
7. Kumar P., Bierlaire M. *Optimizing locations for a vehicle sharing system*: Swiss Transport Research Conference, Ascona, Switzerland, 2012. URL: http://www.strc.ch/conferences/2012/Kumar_Bierlaire.pdf (last access date: 11.11.2020).
8. Patrick Vogel, Torsten Greiser, Dirk Christian Mattfeld. Understanding Bike – Sharing Systems using Data Mining: Exploring Activity Patterns. *Procedia – Social and Behavioral Sciences*. 2011. Vol. 20, No.4, P.514 – 523.
9. Lin, J. R., Yang, T. H., & Chang, Y. C. A hub location inventory model for bicycle sharing system design: Formulation and solution. *Computers & Industrial Engineering*. 2013. Vol. 65, No.1. P. 77 – 86.
10. M. Benchimol, P. Benchimol, B. Chappert, A. de la Taille, F. Laroche, F. Meunier, L. Robinet. Balancing the stations of a self service ‘bike hire’ system. *RAIRO – Operations Research*. 2011. Vol.45, No. 1. P. 37 – 61.

11. D. Chemla, F. Meunier, R.W.Calvo. Bike sharing systems: Solving the static rebalancing problem. *Discrete Optimization*. 2013. Vol.10, No. 2. P. 120 – 146.
12. T. Raviv and O. Kolka. Optimal inventory management of a bike – sharing station, *IIE Transactions*, 2013. Vol. 45, No.10. P. 1077 – 1093.
13. Raviv T., Tzur M., Forma I.A. Static repositioning in a bike – sharing system: models and solution approaches. *EURO Journal on Transportation and Logistics*. 2013. Vol.2, No. 1. P. 187 – 229.
14. Papazek P., Raidl G., Rainer – Harbach M., and Hu B. A PILOT/VND/GRASP hybrid for the static balancing of public bicycle sharing systems, *Computer Aided Systems Theory – EUROCAST*, Springer. 2013. Vol. 8111, No.1. P. 372 – 379.
15. Gaspero Di L., Rendl A., Urli, T. *A hybrid ACO+CP for balancing bicycle sharing systems: Hybrid Metaheuristics 8th International Workshop*, Lecture Notes of Computer Science. Berlin: Springer, 2013. P. 198 – 212.
16. M. Dell'Amico, E. Hadjicostantinou, M. Iori, S. Novellani, The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega*. 2014. Vol.45, No. 1. P. 7 – 19.
17. Erdoğan G, Battarra M, Wolfler Calvo R. An exact algorithm for the static rebalancing problem arising in bicycle sharing systems. *European Journal of Operational Research*. 2015. Vol.245, No. 3. P.667 – 679.
18. Contardo C., Morency C., Rousseau L.M. *Balancing a dynamic public bike – sharing system: Evolutionary Computation in Combinatorial Optimisation*. EvoCOP 2014. Lecture Notes in Computer Science, Berlin, Springer, 2014. P. 73 – 84.
19. Shu, J., Chou, M., Liu, Q., Teo, C. – P., & Wang, I. – L. Bicycle – sharing system: deployment, utilization and the value of re – distribution. *Operations Research*. 2014. Vol. 6, No.61. P. 1346 – 1359.
20. Peiyu Y., Feihu H., Jian P. A Rebalancing Strategy for the Imbalance Problem in Bike – Sharing Systems. *Energies*. 2019. Vol.13, No.12. P.2578 – 2580.
21. Pfrommer J., Warrington J., Schildbach G., Morari M. Dynamic vehicle redistribution and online price incentives in shared mobile systems. *IEEE Transactions on Intelligent Transportation System*. 2014.Vol.15, No.1. P.1567 – 1578.

22. Chiariotti F., Pielli C., Zanella A., Zorzi M. A Dynamic Approach to Rebalancing Bike – Sharing Systems. *Sensors(Basel)*. 2018. Vol.2, No.18. P. 512.
23. F. P. Goksal, I. Karaoglan, and F. Altiparmak. A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery. *Computers & Industrial Engineering*. 2013. Vol. 65, No. 1. P. 39 – 53.

ДОДАТОК А ЛІСТИНГ ОСНОВНОГО МОДУЛЮ ПРОГРАМНОГО ПРОДУКТУ

```

predict.py
import pandas as pd
import numpy as np
from sklearn.preprocessing import
OneHotEncoder
import lightgbm as lgbm
df = pd.read_csv("Station-
StatPerDay_1.txt", sep = ' ', index_col
= "Day").iloc[: -1]
categorical_cols = ["Month", "Weather",
"Season", "DayType"]
numerical_cols = ["Year", "Peak", "Net-
Flow", "Temperature", "Bikes"]
y_col = df["Bikes"].to_numpy()
cat_cols = OneHotEncoder().fit_trans-
form(df[categorical_cols]).toarray()
num_cols = df[numeri-
cal_cols].to_numpy()
df.head()
data_train = np.hstack([cat_cols,
num_cols])
hyper_params = {
    'task': 'train',
    'boosting_type': 'gbdt',
    'objective': 'regression',
    'metric': ['l2', 'auc'],
    'learning_rate': 0.005,
    'feature_fraction': 0.9,
    'bagging_fraction': 0.7,
    'bagging_freq': 10,
    'verbose': 0,
    "max_depth": 8,
    "num_leaves": 128,
    "max_bin": 512,
    "n_estimators": 1000,
    "min_data_in_leaf": 1
}
model = lgbm.LGBMRegressor(**hy-
per_params)
data_train = data_train.astype(float)
y_col = y_col.astype(float)
model.fit(data_train, y_col,
    eval_metric="rmse",
    eval_set=[(data_train, y_col)],
    verbose=True,
    early_stopping_rounds = 10)
repositioning_solver.py
import numpy as np
n = 11
N = [i for i in range(1, n+1)]
V = [0] + N
m = 5
K = [i for i in range(0, m)]

#arcs for each vehicle
A = [(i, j, k) for i in V for j in V for
k in K]

Q = 70
L = [(i, j) for i in K for j in V]
q = [0, 45, 30, 34, 56, -4, 7, -19, -10,
-9, -8, -15]
p = [0, 0, 13, 8, 0, 2, 10, 5, 1, 0, 3,
2]
dataset = open("My.txt", "r")
i = 0
c = {(i, j): 0 for i in range(12) for j
in range(12)}

while True:
    line = dataset.readline()

    if not line:
        break

dataset.close()

from docplex.mp.model import Model
import itertools
f1 = Model('BikeSharingRebalancing-F1')
x = f1.binary_var_dict(A, name='y')
l = f1.continuous_var_dict(L,
name='leverage')

price = {(i): 0 for i in range(5)}
price[0]= 5.156
price[1]=0.33
price[2] = 4.138
price[3] = 5.164
price[4]= 4.727

start = f1.continuous_var_dict(Start,
name='start')

```



```

        f1.add_constraint(f1.sum(x[i,
j, k] for j in S for i in S for k in K) <=
len(S) - f1.max(to_compare, 1))

        mdl.parameters.timelimit = 15
        solutionF1 = f1.solve(log_output=True)
        print(solutionF1)
DPSO
DPSO.h
//
// Created by mariusz on 12.11.18.
//

#ifndef PORR_DPSO_DPSO_H
#define PORR_DPSO_DPSO_H

#include <set>
#include "../graph/Graph.h"
#include "Particle.h"
#include "DPSOConfig.h"

using Swarm = std::vector<Particle>;

class DPSO {
public:
    DPSO(const Graph &graph, NodeID
begin, NodeID end, DPSOConfig config);

    void solve();

    const EdgesSet &get_best_position()
const;

protected:
    virtual void update_best_posi-
tion();

    EdgesSet best_position;
    float best_path_length = std::nu-
meric_limits<float>::max();

private:
    void build_swarm();

    const Graph graph;
    Swarm swarm;
    NodeID begin;
    NodeID end;
    const DPSOConfig config;
};

#endif //PORR_DPSO_DPSO_H

```

```

DPSO.cpp
//
// Created by mariusz on 12.11.18.
//

#include <cassert>
#include <iostream>

#include "../utility/Profiler.h"
#include "DPSO.h"

DPSO::DPSO(const Graph& graph, NodeID
begin, NodeID end, DPSOConfig config)
    : graph(graph), begin(begin), end(end),
config(config) {
    auto validate_endpoints = [&graph](int
id, std::string name) {
        if (id < 0 || id >= graph.size())
            throw std::invalid_argu-
ment(name + " node do not belong to graph");
    };
    validate_endpoints(begin, "Begin");
    validate_endpoints(end, "End");
}

void DPSO::solve() {
    build_swarm();
    update_best_position();

    Profiler& profiler = Profiler::get-
Instance();
    profiler.registerParallelisation-
StartPoint();

    for (auto i = 0; i < config.iterations;
++i) {
        if (config.detailLogs) {
            std::cout << "Itera-
tion=" << i + 1 << "/" << config.iterations <<
" gBest= " << best_path_length << "\n";
        }

#ifdef MODE_OPEN_MP
#pragma omp parallel num_threads(pro-
filer.getThreadNumber())
#pragma omp for
#endif
        for (uint32_t p = 0; p <
swarm.size(); ++p) {
            Particle& particle =
swarm.at(p);
            particle.calculate_ve-
locity(best_position, config);

```

```

        particle.calculate_new_position(config);
        particle.close_new_path(graph, begin, end);
    }
    update_best_position();
}

for (auto p : best_position)
{
    std::cout << p.edge.from << "--
----" << p.edge.to << std::endl;
}
profiler.registerParallelisationStopPoint();
}

void DPSO::build_swarm() {
    assert(swarm.empty());
    for (auto i = 0; i < config.population_cnt; ++i)
        swarm.emplace_back(graph,
begin, end);
}

void DPSO::update_best_position() {
    std::sort(swarm.begin(), swarm.end(),
[] (const auto& lhs, const auto& rhs) {
        return lhs.best_path_length <
rhs.best_path_length;
    });
    auto length = swarm.begin()->best_path_length;
    if (length < best_path_length) {
        best_path_length = length;
        best_position = swarm.begin()->best_position;
    }
}

const EdgesSet& DPSO::get_best_position() const {
    return best_position;
}

DPSOConfig.h
//
// Created by mariusz on 12.11.18.
//

#ifndef PORR_DPSO_DPSOCONFIG_H
#define PORR_DPSO_DPSOCONFIG_H

#include <string>

```

```

struct DPSOConfig {
    int population_cnt = 50;
    int iterations = 150;

    float social_coefficient = 5.0f;
    float cognitive_coefficient = 5.0f;
    float swarm_inertion = 5.0f;
    float previous_pos_impact_coefficient = 5.0f;

    std::string dataset_name = "OL";
    unsigned int number_of_nodes = 30;

    bool detailLogs = false;
};

#endif //PORR_DPSO_DPSOCONFIG_H

DPSOEdge.cpp
//
// Created by mariusz on 12.11.18.
//

#include "DPSOEdge.h"

bool operator==(const DPSOEdge &lhs,
const DPSOEdge &rhs) {
    return lhs.propability == rhs.propability && lhs.edge == rhs.edge;
}

DPSOEdge operator*(DPSOEdge dpsoEdge,
float scalar) {
    dpsoEdge.propability *= scalar;
    if (dpsoEdge.propability > 1.0f)
        dpsoEdge.propability = 1.0f;
    return dpsoEdge;
}

DPSOEdge operator*(float scalar, const
DPSOEdge &dpsoEdge) {
    return dpsoEdge * scalar;
}

DPSOEdge.h
//
// Created by mariusz on 12.11.18.
//

#ifndef PORR_DPSO_DPSOEDGE_H
#define PORR_DPSO_DPSOEDGE_H

```

```

#include "../graph/Edge.h"

struct DPSOEdge {
    Edge edge;
    float propability;

    friend bool operator==(const
DPSOEdge &lhs, const DPSOEdge &rhs);

    friend DPSOEdge operator*(DPSOEdge
dpsOEdge, float scalar);

    friend DPSOEdge operator*(float
scalar, const DPSOEdge &dpsOEdge);
};

namespace std {
    template<>
    struct hash<DPSOEdge> {
        size_t operator()(const
DPSOEdge &obj) const noexcept {
            return
hash<Edge>{}(obj.edge);
        }
    };
}

#endif //PORR_DPSO_DPSOEDGE_H

```

Particle.cpp

```

//
// Created by mariusz on 12.11.18.
//

#include "Particle.h"
#include "DPSOConfig.h"
#include "Astar.h"
#include <random>
//#include <bits/unordered_set.h>
#include <unordered_set>
#include <cassert>
#include <iostream>

bool operator<(const Particle& lhs, const Par-
ticle& rhs) {
    return lhs.best_path_length <
rhs.best_path_length;
}

Particle::Particle(const Graph& graph, const
NodeID from, const NodeID to)

```

```

: position(), best_position(),
best_path_length(std::numeric_lim-
its<float>::max()) {
    /*build_first_path(graph);
    close_new_path(graph, from, to);*/

    int nAvailableVehicles = 3;
    int nCapacity = 5;
    int currPos = 0;
    for (int i = 0; i < graph.size(); ++i)
    {
        int require =
graph.getNode(i).x;
        auto result = require + nCapac-
ity;
        if (!((result < 0 && -result >
nCapacity) || result > nCapacity ))
        {
            auto formEdge =
Edge(currPos, i, 1.0);
            position.insert({
formEdge, 1.0 });
        }
    }

void Particle::build_first_path(const Graph&
graph) {
    std::random_device random_device;
    std::mt19937 generator(random_de-
vice());

    std::uniform_int_distribution<> ran-
dom_node(0, graph.size() - 1);

    for (auto i = 0; i < 10; ++i) {
        auto node_id = static_cast<No-
deID>(random_node(generator));
        const auto& edges = graph.get-
Edges(node_id);

        auto edge_idx = random_node(gen-
erator) % edges.size();
        auto edge = edges[edge_idx];
        if (position.find({ edge, 1.0 })
== position.end())
            position.insert({ edge,
1.0 });
    }

void Particle::update_best_position() {
    auto path_length = calcu-
late_path_length();
    if (path_length < best_path_length) {

```

```

        best_path_length = path_length;
        best_position = position;
    }
}

float Particle::calculate_path_length() {
    float path_length = 0.0f;
    for (auto const& e : position) {
        path_length += e.edge.weight;
    }
    return path_length;
}

void Particle::calculate_velocity(EdgesSet
global_best_position, const DPSOConfig& config)
{
    std::random_device random_device;
    std::mt19937 generator(random_de-
vice());
    std::uniform_real_distribution<> ran-
dom(0.0, 1.0);

    // inertia
    // temporary variable due to const it-
erator in for loop
    EdgesSet new_velocity;
    for (auto& v : velocity)
        new_velocity.insert(v * con-
fig.swarm_inertia);
    velocity = std::move(new_velocity);

    // social
    for (const auto& e : position)
        global_best_position.erase(e);
    for (const auto& e : global_best_posi-
tion)
        velocity.insert(e * config.so-
cial_coefficient * random(generator));

    // cognitive
    auto particle_best_position = best_po-
sition;
    for (const auto& e : position)
        particle_best_posi-
tion.erase(e);
    for (const auto& e : particle_best_po-
sition)
        velocity.insert(e * config.cog-
nitive_coefficient * random(generator));
}

void Particle::calculate_new_position(const
DPSOConfig& config) {
    std::random_device random_device;
    std::mt19937 generator(random_de-
vice());
    std::uniform_real_distribution<> ran-
dom(0.0, 1.0);

    EdgesSet new_position;

    // Filter edges from velocity
    for (const auto& e : velocity) {
        if (random(generator) <= e.prop-
ability)
            new_position.insert(e);
    }

    // Filter edges from current position
    for (const auto& e : position) {
        if (random(generator) <= ran-
dom(generator) * config.previous_pos_im-
pact_coefficient)
            new_position.insert(e);
    }

    // remove duplicates, leaving only the
highest probability
    position.clear();
    if (new_position.empty())
        return;

    auto current_best = *new_posi-
tion.begin();
    static_assert(std::is_same<EdgesSet,
std::unordered_multiset<DPSOEdge>>::value);
    for (const auto& e : new_position) {
        if (current_best.edge == e.edge)
        {
            if (current_best.propaba-
bility < e.propability)
                current_best =
e;
        }
        else {
            position.insert(cur-
rent_best);
            current_best = e;
        }
    }
    if (position.find(current_best) == po-
sition.end())
        position.insert(current_best);
}

std::vector<DPSOEdge> Particle::build_or-
dered_path(const Graph& graph, const NodeID
end) const {

```

```

auto pos = std::vector<DPSOEdge>();
pos.reserve(position.size());
for (const auto& p : position)
    pos.push_back(p);

auto distance_to_end = [&graph,
end](DPSOEdge edge) {
    auto d1 =
graph.straight_line(end, edge.edge.from);
    auto d2 =
graph.straight_line(end, edge.edge.to);
    return d1 < d2 ? d1 : d2;
};

std::sort(pos.begin(), pos.end(),
[&distance_to_end](auto lhs, auto rhs) {
    auto d1 = distance_to_end(lhs);
    auto d2 = distance_to_end(rhs);
    return d1 > d2;
});
return pos;
}

void Particle::close_new_path(const Graph&
graph, const NodeID begin, NodeID end) {
    auto pos = build_ordered_path(graph,
end);

    while (true) {
        auto last_iteration = false;
        auto current_from = end;
        auto current_to = begin;

        if (!pos.empty()) {
            DPSOEdge current =
pos.back();
            pos.pop_back();

            assert(current.edge.from <
static_cast<NodeID>(graph.size()) && "Domain constrain has
been broken");
            assert(current.edge.to <
static_cast<NodeID>(graph.size()) && "Domain
constrain has been broken");
            current_from = cur-
rent.edge.from;
            current_to = cur-
rent.edge.to;
        }
        else {
            last_iteration = true;
        }

        auto path_finder = Astar(graph,
end, current_to);
        auto path_patch =
path_finder.solve();

        for (auto i = 1u; i <
path_patch.size(); ++i) {
            auto from = path_patch[i
- 1];
            auto to = path_patch[i];
            const auto& edges =
graph.getEdges(from);
            auto edge =
std::find_if(edges.begin(), edges.end(),
[to](const Edge& edge) {
                return edge.to
== to;
            });
            auto new_edge =
DPSOEdge{ Edge{from, to, edge->weight}, 1.0 };

            if (posi-
tion.find(new_edge) == position.end())
                position.in-
sert(new_edge);
        }

        if (last_iteration)
            break;

        end = current_from;
    }

    update_best_position();
}

Particle.h

//
// Created by mariusz on 12.11.18.
//

#ifndef PORR_DPSO_PARTICLE_H
#define PORR_DPSO_PARTICLE_H

#include <unordered_set>
#include "DPSOEdge.h"
#include "../graph/Graph.h"
#include "DPSOConfig.h"

using EdgesSet = std::unordered_multi-
set<DPSOEdge>;

```

```

struct Particle {
    friend bool operator<(const Particle &lhs,
const Particle &rhs);

    Particle(const Graph &graph, NodeID from,
NodeID to);

    void update_best_position();

    void calculate_velocity(EdgesSet
global_best_position, const DPSOConfig &con-
fig);

    void calculate_new_position(const DPSOCon-
fig &config);

    void close_new_path(const Graph &graph, No-
deID begin, NodeID end);

    EdgesSet position;
    EdgesSet best_position;
    float best_path_length;
    EdgesSet velocity;

private:
    void build_first_path(const Graph &graph);

    float calculate_path_length();

    std::vector<DPSOEdge> build_or-
dered_path(const Graph &graph, NodeID end)
const;
};

#endif //PORR_DPSO_PARTICLE_H

```