

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

«На правах рукопису»

УДК _____

«До захисту допущено»

Завідувач кафедри

Сергій СТИПЕНКО

(підпис)

(ініціали, прізвище)

“ _____ ” _____ 2022 р.

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-професійною програмою «Інженерія програмного забезпечення
комп'ютерних систем»**

зі спеціальності 121 «Інженерія програмного забезпечення»

**на тему: «Спосіб управління транспортними потоками з використанням
сигнально - інформаційної системи та елементів штучного інтелекту»**

Виконав (-ла):

студент (-ка) VI курсу, групи ІМ-311мп

Казаков Віталій Борисович

Науковий керівник:

Проф. д.т.н Луцький Г.М.

Консультант з нормоконтролю:

проф., д.т.н., Жабін В.І.

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

Студент _____
(підпис)

Київ – 2022 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Рівень вищої освіти – другий (магістерський)

Спеціальність – 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма «Інженерія програмного забезпечення комп'ютерних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Сергій СТИПЕНКО

(підпис) (ініціали, прізвище)

« » _____ 2021 р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Казакову Віталію Борисовичу

1. Тема дисертації «Спосіб управління транспортними потоками з використанням сигнально - інформаційної системи та елементів штучного інтелекту», науковий керівник: Луцький Георгій Михайлович, професор, доктор технічних наук, затверджені наказом по університету від « 4097-с від 08.11.2022»
2. Строк подання студентом дисертації: 20.12.2022
3. Об'єкт дослідження: Процес управління транспортними потоками в інтелектуальній транспортній системі (ITS)
4. Предмет дослідження: Процес використання нелінійних методів аналізу, прогнозування та управління системою керування транспортними потоками з метою зменшення навантажень на транспортні шляхи та запобігання заторам з можливістю застосування на локальному ринку.
5. Перелік завдань, які потрібно розробити: Дослідити основні аспекти регулювання транспортних потоків та вибрати найбільш суттєві. Ознайомитись та визначити інструменти нелінійного аналізу (алгоритми машинного навчання) найбільш продуктивні та запропонувати спосіб використання та комбінації цих інструментів в умовах міста. Визначити можливість реалізації практичного втілення проекту у м. Київ
6. Консультанти розділів дисертації:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	проф., д.т.н., Жабін В.І.		

7. Дата видачі завдання 01.09.2022

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів дисертації	Примітка
1.	Затвердження теми роботи	1.09.2022	
2.	Дослідження літератури та документації	2.09.2022- 21.09.2022	
3.	Огляд існуючих рішень, аналіз теоретичних методів	22.09.2022- 4.10.2022	
4.	Визначення модулів майбутнього продукту	5.10.2022- 10.10.2022	
5.	Збір або аналіз існуючих даних та проведення тренування нейронних мереж	11.10.2022- 23.10.2022	
6.	Проведення аналізу впливу способу управління на ефективність дорожнього руху	24.10.2022- 26.10.2022	
7.	Компонування модулів програмний продукт, тестування.	27.10.2022- 19.11.2022	
8.	Передзахист	30.11.2022	
9.	Захист	22.12.2022	

Студент

(підпис)

В.Б.Казаков

(ініціали, прізвище)

Науковий керівник дисертації

(підпис)

Г.М.Луцький

(ініціали, прізвище)

АНОТАЦІЯ

Магістерська дисертація присвячена вивченню питання управління транспортними потоками шляхом впровадження технологій машинного навчання. Крім того, розглядаються кілька аспектів, які впливають на спосіб управління транспортними потоками. В роботі розглянуто основні підходи, регулювання та моделі управління транспортними потоками. Проведено аналіз існуючих підходів до використання нелінійних методів оцінки та побудови прогнозів і особливості використання алгоритмів Q - навчання (Deep Q-learning), для вирішення питань оптимізації трафіку на перехрестях. Крім того, розглянуто низку алгоритмів аналізу завантаженості шляхів, що включає алгоритми LSTM, GRU, SAEs та інші.

Основним аспектом роботи було створення певного способу управління транспортним потоком завдяки використанню мікро сервісів на базі окремих нейронних мереж, які разом використовуються в одній моделі, але вирішують різні завдання.

Ключові слова: управління транспортними потоками, Q-learning, DQN, SUMO, світлофор

ANNOTATION

The master's thesis is devoted to the study of the issue of traffic flow management through the implementation of machine learning technologies. In addition, several aspects affecting the way of managing traffic flows are considered. The work considers the main approaches, regulation, and models of traffic flow management. An analysis of existing approaches to the use of non-linear methods of assessment and construction of forecasts and features of the use of Q-learning algorithms (Deep Q-learning) to solve issues of traffic optimization at intersections was carried out. In addition, several algorithms for the analysis of path congestion, including LSTM, GRU, SAEs and others, are considered.

The main aspect of the work was the creation of a certain way of managing the traffic flow thanks to the use of micro services based on separate neural networks, which are used together in one model but solve different tasks.

Keywords: traffic flow management, Q-learning, DQN, SUMO, traffic light

РЕФЕРАТ

на магістерську дисертацію

виконану на тему: Спосіб управління транспортними потоками з використанням сигнально - інформаційної системи та елементів штучного інтелекту

студентом: Казаковим Віталієм Борисовичем

Робота складається із вступу та чотирьох розділів. Загальний обсяг роботи: 146 аркуші основного тексту, 45 ілюстрації, 26 таблиць. При підготовці використовувалася література з 27 різних джерел.

Актуальність. Транспортна система, без перебільшення є артерією економічного розвитку, по якій тече потік автотранспорту, який живить розвиток економіки держави. Сучасні тенденції розвитку мегаполісів показують високі темпи росту транспорту у містах при значно повільнішій зміні інфраструктури міст. Великою проблемою сучасності стали транспортні затори, затримки, забруднення повітря, тощо. Ідея створення інтелектуальної транспортної системи виникла мабуть з самого початку появи комп'ютерів і почалась вона з відомої задачі комівояжера. С того часу значно збільшилась кількість задач транспортної системи і відповідно методів їх вирішення – від чисто технічних до сучасних методів прогнозування та управління. Велика кількість досліджень проводиться з метою пошуку оптимальних методів управління світлофорами, планування маршрутів, планування маршрутів доставки товарів, надання інформаційних сервісів. Випробовуються нові моделі машинного навчання та створення штучного інтелекту для управління складними багатовимірними транспортними процесами.

Мета і завдання дослідження. Метою магістерської роботи є зменшення затримки при проходженні перехресть та прогнозування транспортних потоків їх маршрутизація з урахуванням перспектив завантаження. .

Для досягнення мети дослідження поставлено і вирішено такі завдання:

- дослідження структури и поняття інтелектуальної транспортної мережі;
- визначені промислові стандарти проектування та моделювання і симуляції транспортних потоків;
- проаналізовані лінійні та нелінійні алгоритми аналізу, прогнозування та використання їх для управління транспортними потоками в їх реалізації на основних агентах: світлофорах, інформаційних каналах та системах інформування;
- ілюстрація роботи моделі та аналіз отриманих результатів

Об’єкт дослідження – процес управління транспортними потоками в інтелектуальній транспортній системі (ITS).

Предмет дослідження – процес використання нелінійних методів аналізу, прогнозування та управління системою керування транспортними потоками з метою зменшення навантажень на транспортні шляхи та запобігання заторам з можливістю застосування на локальному ринку.

Методи досліджень. Для досягнення поставлених в магістерській роботі задач, використано багаторівнева система мікро сервісів з застосуванням нелінійних методів аналізу (нейромереж) для системи управління світлофорами та прогнозування транспортних потоків.

Наукова новизна одержаних результатів роботи полягає у наступному:

- Запропоновано багаторівневий мікро сервісний продукт управління транспортним потоком, який можливо реалізувати в локальному середовищі;
- Розроблено моделі нелінійного аналізу та проведено симуляцію роботи алгоритмів в SUMO.
- Запропоновано використання способу у пілотному проекті або симуляції на реальних даних м. Київ

Проведена робота може допомогти в зборі первинної інформації щодо дорожнього руху у м. Київ та налаштування способу на реальні дані локальних агентів (світлофорів).

Особистий внесок здобувача. Магістерська робота була проведена

самостійно, базувалась на великій кількості первинних матеріалів та досліджень у галузі управління транспортними потоками та досліджень, щодо застосування нелінійних методів аналізу даних, управління та провадження методів машинного навчання в систему інтелектуального управління транспортом.

Практична цінність. Отримані результати можуть використовуватися у практичному провадженні та теоретичних дослідженнях на локальному ринку:

- Збір первинної інформації щодо транспортних потоків;
- Аналіз ефективності застосування алгоритмів на реальних даних шляхом пілотних проектів;
- Доопрацювання моделі до комерційного локального модуля керування світлофорами.

Публікації:

Ключові слова

Інтелектуальна транспортна система, машинне навчання, DQN, SUMO, світлофор.

Пояснювальна записка

до магістерської дисертації

на тему: «Спосіб управління транспортними потоками з використанням сигнально- інформаційної системи та елементів штучного інтелекту»

Київ – 2022

ЗМІСТ

РЕФЕРАТ	5
на магістерську дисертацію	5
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І	
ТЕРМІНІВ	13
ВСТУП.....	14
РОЗДІЛ 1	15
ДОСЛІДЖЕННЯ ПРИНЦИПІВ ПОБУДОВИ ІНТЕЛЕКТУАЛЬНОЇ	
ТРАНСПОРТНОЇ СИСТЕМИ	15
1.1. Структура ІТС	15
1.2. Компоненти архітектури ІТС та їх застосування	20
1.3. Функція ІТС – управління трафіком.....	24
1.4. Основні компоненти для аналізу та моделювання.....	37
1.4.1. Система світлофорів	37
1.4.2. Швидкісні смуги	38
1.4.3. Планування маршруту.....	39
1.4.4. Уникнення заторів у міському русі.....	41
1.4.5. Децентралізована передбачувана маршрутизація	43
1.5. Інструменти моделювання та симуляції архітектури ІТС	43
1.5.1. Основні компоненти моделювання.....	46
1.5.2. Джерела інформації для системи ІТС.....	47
Висновки до розділу 1.....	50
РОЗДІЛ 2.....	51
МОДЕЛІ АНАЛІЗА ТА УПРАВЛІННЯ ДОРОЖНІМ ТРАФИКОМ З	
ДОПОМОГОЮ НЕЙРОМЕРЕЖ.....	51
2.1. Категоризація моделей транспортних потоків	51

	10
2.1.1. Моделювання транспортних потоків.....	51
2.1.2. Рівень опрацьованості деталей трафіку.....	52
2.1.3. Прогнозування трафіку	54
2.1.4. Прогнозування заторів	56
2.1.5. Типи даних та джерела.....	57
2.2. Алгоритми аналізу транспортних потоків	59
2.2.1. Алгоритми кластеризації	59
2.2.2. Статистичний підхід.....	59
2.2.3. Підхід до машинного навчання	60
2.2.4. Підхід до глибокого навчання.....	61
2.2.5. Як реалізувати прогнозування трафіку	63
2.2.6. Готові рішення	63
2.2.7. Використання нелінійних методів аналізу та прогнозування трафіку.....	64
2.2.8. Глибокі нейронні мережі (DNN)	66
2.2.9. CNN (згорточні нейронні мережі).....	67
2.2.10. SAE мережі (автоенкодери).....	68
2.2.11. RNN мережі	68
2.2.12. Гібридні мережі.....	69
Висновки до розділу 2	73
РОЗДІЛ 3	74
РЕАЛІЗАЦІЯ СПОСОБУ УПРАВЛІННЯ ТРАНСПОРТНИМ ПОТОКОМ З ВИКОРИСТАННЯМ НЕЙРОМЕРЕЖІ	74
3.1. Опис запропонованого способу управління транспортним потоком	

3.2. Реалізація способу управління потоками за допомогою світлофорів.....	78
3.2.1. Спосіб реалізації на базі локального агента (світлофору) – управління на базі визначення завантаження перехрестя	78
3.2.2. Спосіб реалізації на базі RL (reinforcement learning) (світлофору)	82
3.2.2.1. DQN використання в програмі	83
3.3. Спосіб управління потоками за рахунок прогнозування транспортних потоків та їх перенаправлення	90
3.4. Моделювання та симуляція алгоритму прогнозування та моделювання транспортного потоку	92
3.4.1. Результати використання способу управління транспортним потоком	96
Висновки до розділу 3	98
РОЗДІЛ 4.....	99
РОЗРОБКА СТАРТ АП ПРОЕКТУ.....	99
4.1. Опис проекту	99
4.2. Порівняльний аналіз конкурентних систем	101
4.3. Оцінка ідеї проекту.....	103
4.4. Аналіз оцінки можливості провадження проекту на ринку	104
4.5. Розроблення ринкової стратегії.....	114
4.6. Розроблення маркетингової програми проекту	118
Висновки до розділу 4	122
ВИСНОВКИ	123
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	125
Додаток А	128
Додаток Б	129

Додаток В 130

Додаток Г 131

Додаток Д.1 132

Додаток Д.2 133

Додаток Д.3 133

Додаток Д.4 134

Додаток Д.5 135

Додаток Е 136

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

ITS (ITC) – Інтелектуальна транспортна система

DQN (Deep Q Network) – глибока Q мережа

LSTM (Long Short-Term Memory) – мережа довгострокової - короткострокової пам'яті

GRU (Gated Recurrent Unit) – керований рекурентний блок

SAEs (Sparse Autoencoder) – автоенкодер розріджений

TMS (Transport Management System)– система транспортного менеджменту

ISO – міжнародна організація стандартів

RF (Random Forest) – метод машинного навчання випадковий ліс

SVR (Support Vector Regression) – метод опорних векторів

ARIMA (Autoregressive Integrated Moving Average) - інтегрована модель авторегресії - ковзного середнього

ML (Machine Learning) – машинне навчання

KNN (K – Nearest Neighbor) - метод K – найближчих сусідів

DL (Deep Learning) – глибоке навчання

RL (Reinforcement Learning) – навчання з підкріпленням

YOLO (You Only Look Once) – нейронна мережа визначення об'єктів

SGD (Stochastic Gradient Decent) – стохастичний градієнтний спуск

RMSprop (Root Mean Square Propagation) – метод зворотного розповсюдження помилки

SUMO(Simulation of Urban Mobility) – симулятор міського трафіку

ВСТУП

Прогнозування транспортних потоків є ключовою частиною інтелектуальної транспортної системи. Прогнозування передбачає точну оцінку транспортного потоку у заданому регіоні у певний інтервал часу у майбутньому. Вивчення прогнозування дорожнього руху корисне для зменшення заторів та забезпечення більш безпечної та економічної подорожі.

Традиційні моделі прогнозування використовують неглибокі нейронні мережі, однак збільшення факторів впливу та агентів - учасників руху призвело до зменшення ваги традиційних моделей машинного навчання.

Кількість транспортних засобів постійно збільшується, транспортна інфраструктура не встигає за розвитком ситуації, що провокує затори, які є серйозною проблемою, а у випадку дорожньо-транспортних пригод вони стають ще серйознішими. Ця проблема торкається багатьох аспектів сучасного суспільства, включаючи економічний розвиток, дорожньо-транспортні пригоди, збільшення викидів парникових газів, час і проблеми зі здоров'ям. У цьому контексті сучасні суспільства можуть покладатися на систему управління дорожнім рухом, щоб мінімізувати затори на дорогах та їх негативні наслідки. Для усунення цього ефекту було запропоновано спосіб управління дорожнім рухом (TMS) з використання машинного навчання. Ця система орієнтована на моніторинг транспортних засобів, щоб скоротити час, що проводиться на світлофорах, виявляти та запобігати заторам на дорогах та пропонувати альтернативні маршрути для транспортних засобів. В даний час транспортний потік і затори є однією з основних соціальних та економічних проблем, пов'язаних із транспортом у промислово розвинених країнах. У цьому плані управління трафіком в перенавантажених мережах вимагає чіткого розуміння операцій із потоком трафіку. Тобто розуміння того, що викликає перевантаження, що визначає час і місце збою трафіку, як перевантаження поширюється мережею має важливе значення.

РОЗДІЛ 1

ДОСЛІДЖЕННЯ ПРИНЦИПІВ ПОБУДОВИ ІНТЕЛЕКТУАЛЬНОЇ ТРАНСПОРТНОЇ СИСТЕМИ

1.1. Структура ІТС

Інтелектуальні транспортні системи (ІТС) - це назва транспортної системи, в яких транспортні засоби взаємодіють з навколишнім середовищем та один з одним для забезпечення більш комфортного водіння, а інтелектуальна інфраструктура підвищує безпеку та пропускну спроможність дорожніх систем.

Інтелектуальні транспортні системи (ІТС) - це назва, це набір складних і сучасних систем, покликаних поліпшити враження від водіння. У цих системах беруть участь транспортні засоби, водії, пасажери, дорожні оператори та менеджери, що взаємодіють один з одним та з навколишнім середовищем та пов'язані найчастіше зі складними магістральними інфраструктурними системами.

Національна архітектура ITS забезпечує загальну основу для планування, визначення та інтеграції інтелектуальних транспортних систем. Це продукт, який відображає внесок широкого перерізу спільноти ITS (практиків транспорту, системних інженерів, розробників систем, спеціалістів з технологій, консультантів тощо).

Архітектура визначає:

Функції (наприклад, збір інформації про дорожній рух або запит маршруту), необхідні для ITS.

Фізичні об'єкти або підсистеми, де знаходяться ці функції (наприклад, поле або транспортний засіб).

Потоки інформації та даних, які з'єднують ці функції та фізичні підсистеми в інтегровану систему.[15]

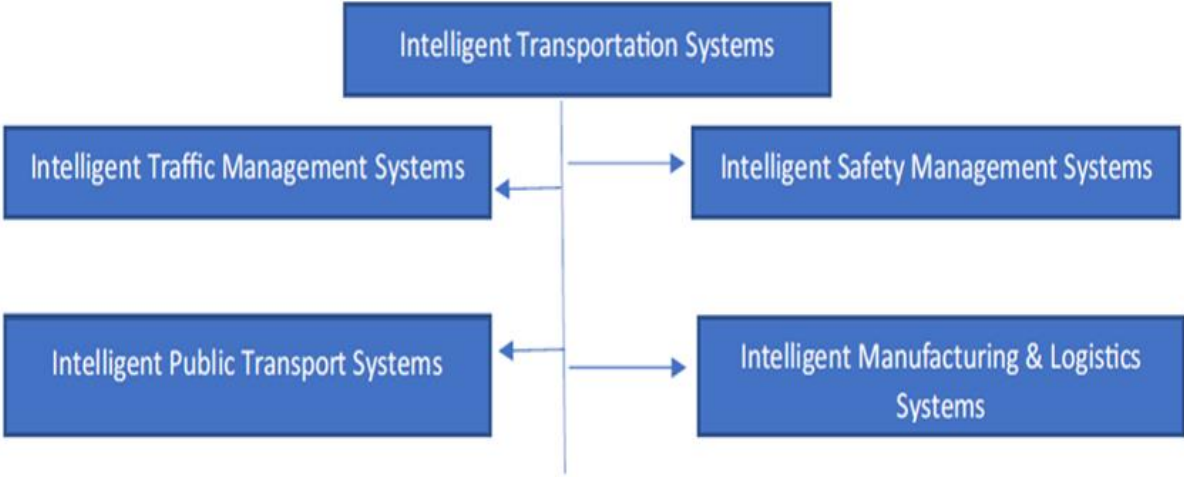


Рис.1.1.Узагальнена архітектура інтелектуальної транспортної системи [15]

Ключові компоненти національної архітектури ІТС підсумовано нижче:

Таблиця 1.1. Компоненти ІТС

Подорожі та управління дорожнім рухом

Послуги	
1.1	Інформація про поїздку перед поїздкою
1.2	Інформація про водія на маршруті
1.3	Прокладка маршруту
1.4	Підбір поїздок і бронювання
1.5	Інформація про послуги для подорожуючих
1.6	Контроль дорожнього руху
1.7	Управління інцидентами
1.8	Управління попитом на подорожі
1.9	Випробування викидів і пом'якшення
1.10	Залізничний перетин шосе

Таблиця 1.1. Компоненти ІТС (продовження)

Управління публічним транспортом

Послуги	
2.1	Управління громадським транспортом
2.2	Інформація про транзит на маршруті
2.3	Персоналізований громадський транспорт
2.4	Безпека громадського транспорту

Електроні платежі

Послуги	
3.1	Послуги електронних платежів

Експлуатація комерційних транспортних засобів

Послуги	
4.1	Електронне оформлення комерційного автомобіля
4.2	Автоматизована система безпеки на дорозі
4.3	Бортова система безпеки та моніторинг безпеки
4.4	Адміністративні процеси комерційних автомобілів
4.5	Безпека небезпечних матеріалів та реагування на інциденти
4.6	Мобільність вантажів

Управління надзвичайними ситуаціями

Послуги	
5.1	Екстрене сповіщення та особиста безпека
5.2	Управління екстремим транспортним засобом
5.3	Ліквідація та евакуація

Таблиця 1.1. Компоненти ІТС (продовження)

Передові системи безпеки автомобіля

Послуги	
6.1	Уникнення поздовжнього зіткнення
6.2	Уникнення бічного зіткнення
6.3	Уникнення зіткнення на перехресті
6.4	Покращення зору для запобігання ДТП
6.5	Готовність до безпеки
6.6	Розгортання передаварійної системи обмеження
6.7	Автоматизована робота автомобіля

Управління інформацією

Послуги	
7.1	Функція архівації даних

Технічне обслуговування та будівництво

Послуги	
8.1	Технічне обслуговування та будівництво

ISO 14813-1:2015 ідентифікує такі «службові домени»:

- Інформація для мандрівників (подорожніх)
- Управління трафіком та операції
- Автосервіс
- Вантажний транспорт
- Громадський транспорт
- Надзвичайна ситуація
- Електронний платіж на транспорті
- Особиста безпека, пов'язана з дорожнім транспортом
- Моніторинг погоди та умов навколишнього середовища
- Управління реагуванням на лиха та координація
- Національна безпека

На рисунку нижче представлена фізична структура ІТС

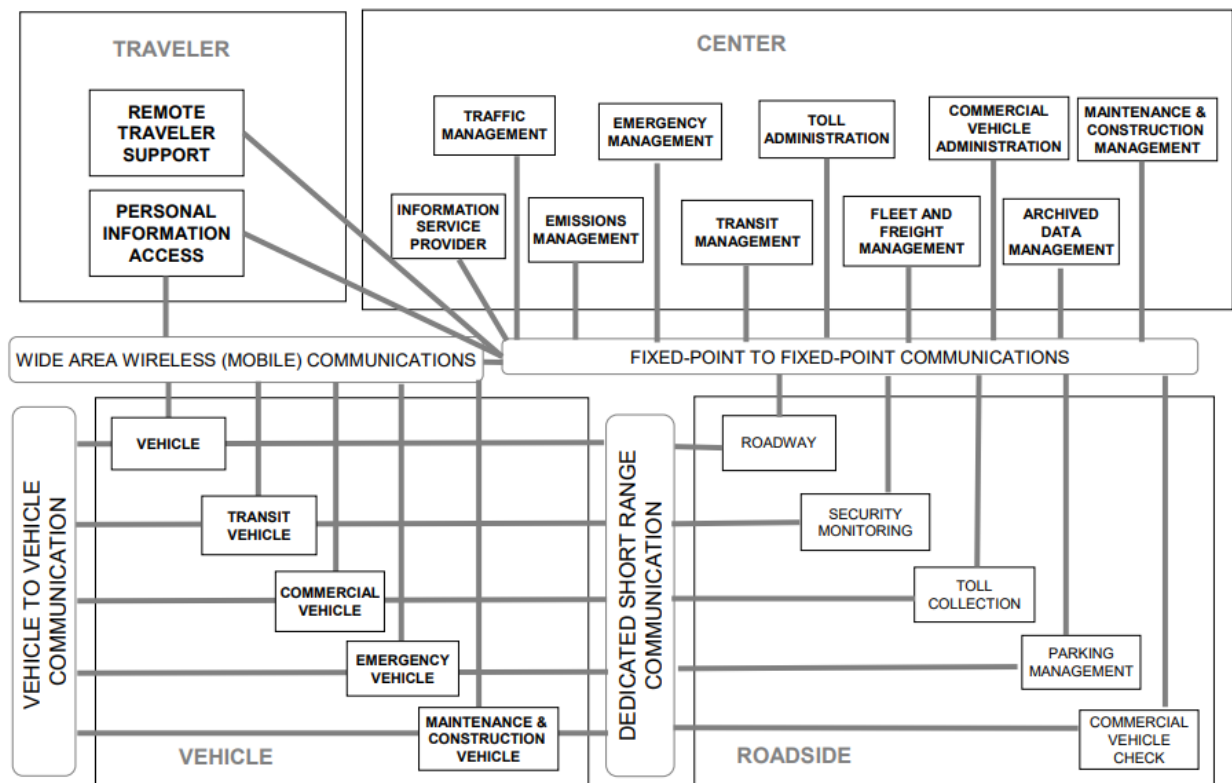


Рис.1.2. Фізична структура ІТС мережі [1]

Хоча ІТС часто розглядається в контексті дорожнього руху, спільне використання видів транспорту пасажирями та вантажними транспортними засобами означає, що ІТС також включає інтерфейс із залізничними, водними та повітряними транспортними системами.

Системи ІТС не працюють ізольовано. Вони часто залежать від магістральних інфраструктурних систем для телекомунікацій, інфраструктурних мереж та Інтернету.

Інтелектуальні транспортні системи не обов'язково мають бути пов'язані лише з транспортними засобами та дорогами. Повітряний транспорт, морський транспорт та залізничні транспортні системи можуть ставати все більш інтелектуальними. Повітряні та залізничні транспортні системи використовували передові системи та електроніку як частину своєї роботи та інфраструктури. Морські навігаційні системи для всіх суден також тривалий час використовували електроніку та радіо для визначення місцезнаходження, обходу перешкод та запобігання зіткненням.

ISO TC204 - Комітет Міжнародної організації зі стандартизації стандартів ІТС - визначає ІТС як «інформаційні, комунікаційні та керуючі системи в галузі міського та сільського наземного транспорту, включаючи інтермодальні та мультимодальні аспекти, інформацію для мандрівників, керування дорожнім рухом, громадський транспорт, комерційний транспорт, екстрені служби та комерційні служби, які зазвичай називають «інтелектуальними транспортними системами (ІТС)».

1.2. Компоненти архітектури ІТС та їх застосування

Інформація для мандрівників (подорожніх)

Міжнародний стандарт ISO 14813-1 описує цей домен як «Надання користувачам як статичної, так і динамічної інформації про транспортну мережу, включаючи модальні опції та передачі».

Інформаційна область мандрівника включає групи послуг:

- Інформація перед поїздкою
- Інформація про подорож
- Керівництво за маршрутом та навігація перед поїздкою
- Керівництво за маршрутом та навігація під час поїздки
- Підтримка планування поїздки
- Інформація про туристичні послуги

Управління трафіком

Міжнародний стандарт ISO 14813-1 описує цю область як «Управління рухом транспортних засобів, мандрівників та пішоходів мережею автомобільного транспорту».

Домен управління трафіком і операцій включає сервісні групи:

- Управління та контроль трафіку
- Управління інцидентами, пов'язаними з транспортом

- Управління попитом
- Управління обслуговуванням транспортної інфраструктури
- Охорона/дотримання правил дорожнього руху

Автомобільні послуги

Міжнародний стандарт ISO 14813-1 описує цю сферу як «Підвищення безпеки, захищеності та ефективності експлуатації транспортних засобів за рахунок попереджень та допомоги користувачам або управління операціями транспортних засобів».

Автосервіс включає групи послуг:

- Поліпшення зору, пов'язане з транспортом
- Автоматизована робота транспортного засобу
- Уникнення зіткнення
- Готовність до безпеки
- Розгортання стримувальних пристроїв перед зіткненням

Вантажний транспорт та логістика

Міжнародний стандарт ISO 14813-1 описує цю сферу як «Управління операціями з комерційними транспортними засобами; управління вантажами та флотом; дії, які прискорюють процес авторизації вантажів на національних та юрисдикційних кордонах та прискорюють крос-модальні перевезення дозволених вантажів».

Громадський транспорт

Міжнародний стандарт ISO 14813-1 описує цю сферу як «Експлуатація послуг громадського транспорту та надання оперативної інформації оператору та користувачу, включаючи мультимодальні аспекти».

Домен громадського транспорту включає групи послуг:

Управління громадським транспортом

Реагує на попит і транспорт, що спільно використовується

Надзвичайна ситуація

Міжнародний стандарт ISO 14813-1 описує цю сферу як «Послуги, що надаються у відповідь на інциденти, які класифікуються як надзвичайні ситуації».

Аварійний домен включає сервісні групи:

- Оповіщення про надзвичайні ситуації на транспорті та особиста безпека
- Відновлення автомобіля після викрадення
- Управління аварійним транспортом
- Запобігання аварійному автомобілю
- Дані аварійного автомобіля
- Небезпечні матеріали та повідомлення про інциденти

Електронний платіж на транспорті

Міжнародний стандарт ISO 14813-1 описує цей домен як «Транзакції та резервування транспортних послуг».

Домен електронних платежів, пов'язаних з транспортом, включає групи послуг:

- Електронні фінансові операції, пов'язані з транспортом
- Інтеграція транспортних електронних платіжних сервісів

Особиста безпека, пов'язана з дорожнім транспортом

Міжнародний стандарт ISO 14813-1 описує цю область як «Захист користувачів транспорту, включаючи пішоходів та вразливих користувачів. “

Безпека на автомобільному транспорті включає групи послуг:

- Безпека громадського транспорту
- Підвищення безпеки для вразливих учасників дорожнього руху
- Підвищення безпеки для інвалідів-учасників дорожнього руху
- Заходи безпеки для пішоходів під час використання інтелектуальних перехресть та переходів

Моніторинг погоди та умов навколишнього середовища

Міжнародний стандарт ISO 14813-1 описує цей домен як «Діяльність, яка відстежує та повідомляє про погодні умови та умови довкілля. “

Домен «Моніторинг погодних та екологічних умов» включає сервісні групи:

Моніторинг умов довкілля

Управління реагуванням на лиха та координація

Міжнародний стандарт ISO 14813-1 описує цю сферу як «Дії на автомобільному транспорті у відповідь на стихійні лиха, громадянські заворушення або терористичні акти».

Управління та координація реагування на стихійні лиха включає групи обслуговування:

- Управління даними про стихійні лиха
- Управління реагуванням на лиха
- Координація з аварійними службами

Національна безпека

Міжнародний стандарт ISO 14813-1 описує цю сферу як «Діяльність, яка безпосередньо захищає або пом'якшує фізичні чи експлуатаційні збитки особам та об'єктам внаслідок стихійних лих, цивільних заворушень або терористичних атак».

Домен національної безпеки включає сервісні групи:

- Моніторинг та контроль підозрілих транспортних засобів
- Моніторинг інженерних мереж чи трубопроводів

Управління даними ІТС

Міжнародний стандарт ISO 14813-1 описує цю сферу як «співставлення, управління та надання даних ІТС законним зацікавленим сторонам».

Домен управління даними ІТС включає сервісні групи:

- Реєстри даних
- Словники даних
- Екстрені повідомлення
- Дані центру управління
- Правозастосування
- Дані управління трафіком

1.3. Функція ІТС – управління трафіком

Метою управління та контролю дорожнього руху є забезпечення безпечного руху користувачів дорожніх транспортних засобів.

Першим інструментом управління дорожнім рухом стала розділена проїжджа частина з відповідним правилом завжди рухатись ліворуч або праворуч від розділової лінії. Першим інструментом ІТС був світлофор із відповідним правилом зупинки на червоне світло. Проте з комп'ютеризацією, створенням мереж, детекторними системами, набір інструментів, що використовуються для управління транспортом, значно збільшився та продовжує розширюватися, особливо у сфері прямої взаємодії з учасниками дорожнього руху через спільні ІТС.

Стандарти управління та контролю трафіку представлені в наступних функціональних областях:

- Архітектури управління та контролю трафіку
- Виявлення та ідентифікація учасників дорожнього руху
- Робота зі світлофорами
- Контроль доступу до зони
- Контроль швидкості руху
- Моделювання трафіку
- Управління дорожніми пригодами
- Управління дорожніми роботами

- Управління якістю повітря
- Інформація про дорожню мережу та поради
- Оптимізація дорожньої мережі
- Спеціальне керування маршрутами користувачів
- Забезпечення дотримання правил дорожнього руху

Архітектури управління та контролю трафіку

Архітектури управління та контролю трафіку

Метою цієї функціональної області є інтеграція систем та інструментів, необхідних для забезпечення ефективного керування та контролю трафіку.

Управління дорожнім рухом є складною функціональною областю, що охоплює ряд механізмів від регулювання та інфраструктури до тактичного контролю.

На рис.1.3 показано, загальну архітектуру управління трафіком за ІТС [1].

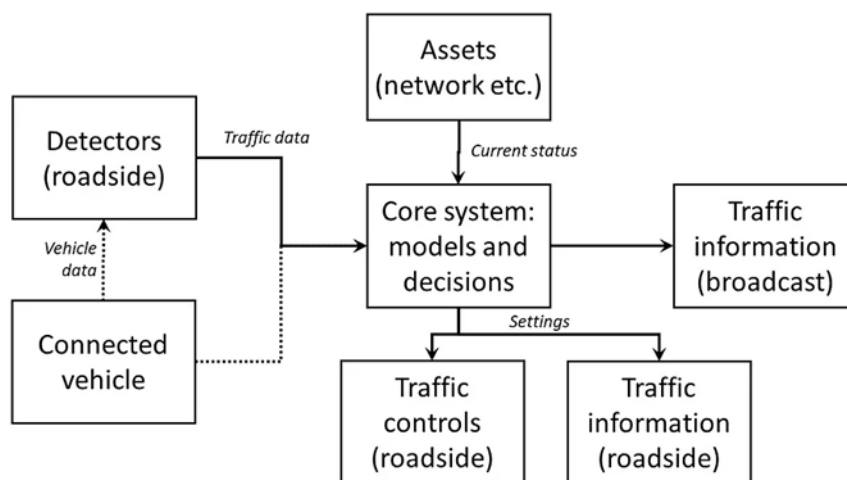


Рис 1.3. Архітектура систем високого рівня управління трафіком[1]

Таким чином, інструменти ІТС у цій прикладній галузі повинні:

- узгоджувати та підтримувати політичний вибір та рішення
- враховувати фіксовані характеристики конструкції дорожньої мережі та пов'язаності
- враховувати фіксовані втручання, такі як жорсткі показники та дорожня розмітка.

Виявлення та ідентифікація учасників дорожнього руху

Метою цієї функціональної області є моніторинг того, хто використовує дорожню мережу, для інформування про управління дорожнім рухом, контроль та інформаційні стратегії.

Управління дорожнім рухом може здійснюватися пасивно (наприклад, просто з використанням фіксованих знаків обмеження швидкості, обмежень повороту тощо), і навіть ІТС не завжди повинні бути «інтелектуальними» (наприклад, світлофори, як і раніше, виконують корисну функцію безпеки, коли вони працюють за фіксованим графіком). Однак у багатьох випадках, особливо при інтенсивному русі, зміні протягом дня та наявності різних учасників дорожнього руху з різними політичними пріоритетами, дуже корисно відстежувати поточний стан дорожньої мережі за допомогою виявлення та/або ідентифікації транспортних засобів.

Отримані таким чином дані можуть дати інформацію про поточні та майбутні потоки, яку можна порівняти з моделями пропускної спроможності доріг для прогнозування потенційних проблем із заторами. Дані можуть також вказувати на фактичні або ймовірні затори. Це може дозволити ввести зміни до управління ІТС (наприклад, налаштування сигналів світлофора, обмеження швидкості, вказівки маршруту або інформаційні повідомлення).

Робота зі світлофорами

Призначення цієї функціональної області – керувати налаштуваннями сигналів світлофора, щоб підтримувати транспортний потік для учасників дорожнього руху.

Сигнали світлофора є основним заходом контролю. Вони налаштовані працювати на перехресті, пішохідному переході, щоб або пропускати, або зупиняти рух транспорту за певним маршрутом.

Системне середовище для роботи світлофорів включає самі світлофори, контролери сигналів: один контролер зазвичай відповідає лише за сигнали

навколо певної точки мережі (наприклад, перехрестя), але іноді його також можна налаштувати для керування іншими сусідніми сигналами.

На рисунку 1.4. Зображено систему керування сигналами світлофору[1].

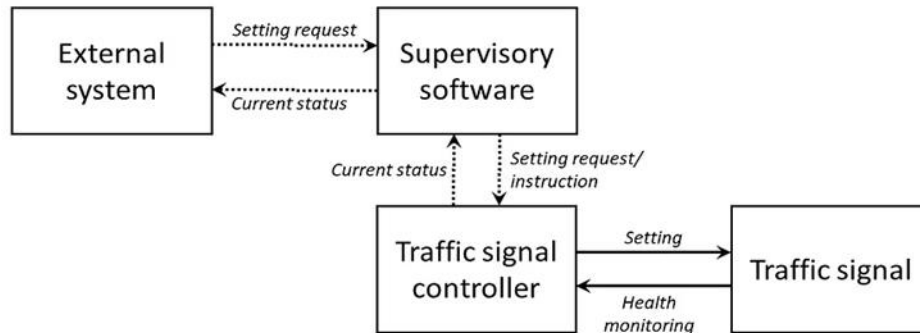


Рис 1.4. Системи керування сигналами світлофора[1]

Системний підхід, використовуваний програмним забезпеченням для диспетчерського управління, дуже різноманітний. Деякі рішення досить локалізовані і навіть можуть бути вбудовані в контролер сигналів; наприклад, продукти, що базуються на алгоритмах MOVA (мікропроцесорна активація автомобіля), використовують дані від локальних детекторів для налаштування параметрів сигналу в режимі реального часу. Інші працюють на вищому рівні та можуть бути ієрархічними; наприклад, рішення, засновані на архітектурі UTMC, можуть мати стратегічні інструменти.

Контроль доступу до зони

Метою цієї функціональної області є керування проходженням трафіку з однієї частини мережі до іншої або між мережами, де діють особливі умови.

Політика дорожнього руху часто включає вимогу обмежити чи заборонити певним типам учасників дорожнього руху використання певних частин мережі. Такі обмеження може бути кількох типів.

Деякі дороги можуть бути не в змозі витримати певні типи транспортних засобів - наприклад, великовантажним автомобілям може знадобитися

триматися подалі від вузьких провулків або слабких мостів, а велосипедам може знадобитися триматися подалі від дуже швидких доріг.

Деякі зони можуть підлягати екологічному чи культурному контролю – наприклад, частковий чи весь моторизований рух може бути обмежений з історичних районів міста чи зон, де якість повітря є особливою проблемою.

Деякі ділянки доріг можуть бути призначені лише для певних транспортних засобів у рамках транспортної політики, наприклад, дороги лише для автобусів.

На деяких ділянках доріг може бути встановлений строгий контроль безпеки, наприклад, на дорогах поряд з будівлями, що охороняються.

На деяких дорогах можуть бути обмеження руху через моделювання, яке показує, що загальний транспортний потік покращується завдяки певній геометрії мережі.

Контроль швидкості руху

Метою цієї функціональної області є контроль швидкості руху за допомогою обов'язкових чи рекомендаційних обмежень швидкості.

Постійні обмеження швидкості на ділянках доріг будуть визначатися рядом факторів, включаючи національне законодавство (залежно від типу дороги), рішення місцевої політики (як правило, які визначаються для забезпечення максимальної безпеки та/або мережного потоку). Крім того, обмеження швидкості в деяких місцях можуть бути тимчасово знижені залежно від поточних умов завантаженості мережі, погоди, подій тощо.

Там, де обмеження швидкості є динамічними знадобиться компонент ITS. Типовим рішенням ITS для динамічних обмежень швидкості є табло зі змінним повідомленням, яке імітує знак фіксованого обмеження швидкості, але зображення якого може бути змінено відповідно до поточних вимог управління.

Обмеження швидкості, що застосовуються до транспортних засобів, а не сегментів доріг, рідко викликають занепокоєння органів дорожнього руху. Звичайними технологіями, що використовуються в цьому випадку, є тахографи

та обмежувачі швидкості транспортних засобів, що повністю вбудовані в структуру транспортного засобу (або іноді мають зв'язок у реальному часі з менеджером автопарку).

Однак і тут є актуальною технологія «Інтелектуальної адаптації швидкості» (ISA). Відповідно до ISA зовнішній орган має право накладати обмеження швидкості безпосередньо на транспортні засоби, що перебувають під його контролем; в принципі це може бути динамічним і навіть не потрібно інформувати користувача транспортного засобу (наприклад, це може бути доречним для автоматизованих транспортних засобів). Очевидно, що ISA вимагає як транспортного засобу, здатного до зовнішнього обмеження швидкості, так і механізму C-ITS для повідомлення поточних обмежень швидкості.

Моделювання трафіку

Метою цієї функціональної області є оцінка розвитку поведінки трафіку з урахуванням доступних поточних даних.

Моделювання - це широка дисципліна, і існує безліч підходів, доступних для конкретних обставин:

На найвищому рівні існують такі методи як гравітаційна модель, яка прагне зрозуміти довгостроковий розвиток попиту на перевезення між населеними пунктами на основі основних демографічних факторів чисельності населення та фізичної відстані.

У середині є такі методи, як моделі призначення, в яких попит на поїздки з пункту відправлення до пункту призначення відображається на графі мережі вузлів і каналів; варіанти можуть включати чи виключати вибір режиму, час доби, транспортні витрати тощо.

Більше деталізованими є інструменти мікро моделювання, в яких окремим учасникам дорожнього руху задається поведінка, і спостерігається результуючий вплив на мережу, особливо для точок мережі, де неодноразово виникають затори, але іноді також і для інших проблем, таких як ймовірність аварії.

Існують докладні моделі індивідуальної поведінки при водінні, засновані на таких людських факторах, як втома, уважність, стрес, знання маршруту тощо. Вони можуть поєднуватися з локальними моделями проектування перехресть.

Моделювання трафіку в більшості випадків здійснюється в автономному режимі: дані збираються, відповідна модель запускається (один або кілька разів), результати моделі документуються, і готується звіт, що фіксує результати процесу і тим самим допомагає у прийнятті рішень.

Це стосується моделювання мережових інвестицій та розвитку.

Виняток становлять моделі, що інтегруються в оперативний контроль. Наприклад, в якійсь точці мережі відбувається аварія, і орган керування дорожнім рухом повинен знати, як налаштувати свої служби керування трафіком та інформаційні послуги, щоб звести до мінімуму наслідки.

Якщо оперативна інформація подається в модель оптимальне рішення може бути швидко досягнуто. Це може бути додатково автоматизовано з автоматичним поверненням оптимального рішення для налаштування параметрів сигналу і т. д. Цей підхід, однак, надзвичайно складний і технічно складний, і в даний час він не набув широкого поширення.

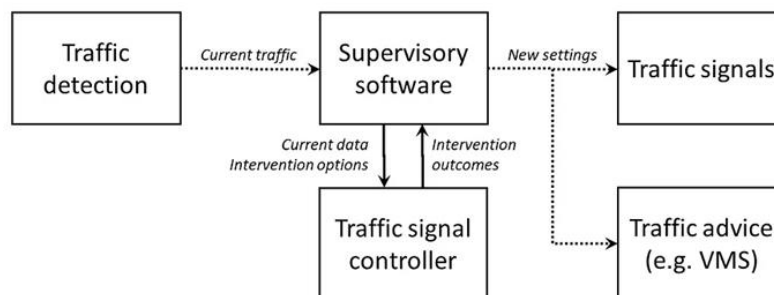


Рис 1.5. Системи оперативного вибору моделі керування[1]

На рисунку представлений процес вибору моделі керування, як результат реакції системи на події.

Управління дорожніми пригодами

Метою цієї функціональної області є виявлення та усунення незапланованих збоїв у дорожній мережі.

Виявлення може відбуватися в результаті прямого спостереження, зовнішнього звіту або виведення з іншої інформації моніторингу. Оскільки подія може розвиватися непередбачуваним чином, збір аналогічної інформації часто необхідний протягом усього процесу управління пригодою, враховуючи, що, як стане відомо про подію, диспетчерам дорожнього руху та учасникам дорожнього руху, буде запропоновано шукати поточну інформацію.

Інструменти, доступні безпосередньо диспетчеру трафіку, включають: системи управління інцидентами, технології, що документують описи інциденту, вжитих дій та поточної ситуації.

Втручання, не пов'язані з ІТС, такі як тимчасове закриття дороги фізичними бар'єрами, наприклад, для проведення ремонтних робіт у дорожній інфраструктурі.

Надання інформації безпосередньо поточним учасникам дорожнього руху через VMS та потенційним учасникам дорожнього руху.

Надання інформації учасникам дорожнього руху опосередковано через сторонніх постачальників інформаційних.

Надання інформації керівникам транспортних компаній (наприклад, операторам громадського транспорту або вантажним операторам), які потім можуть контролювати поведінку (маршрут, швидкість, час тощо) транспортних засобів, що знаходяться в їхньому управлінні. На малюнку нижче показаний система управління інцидентами

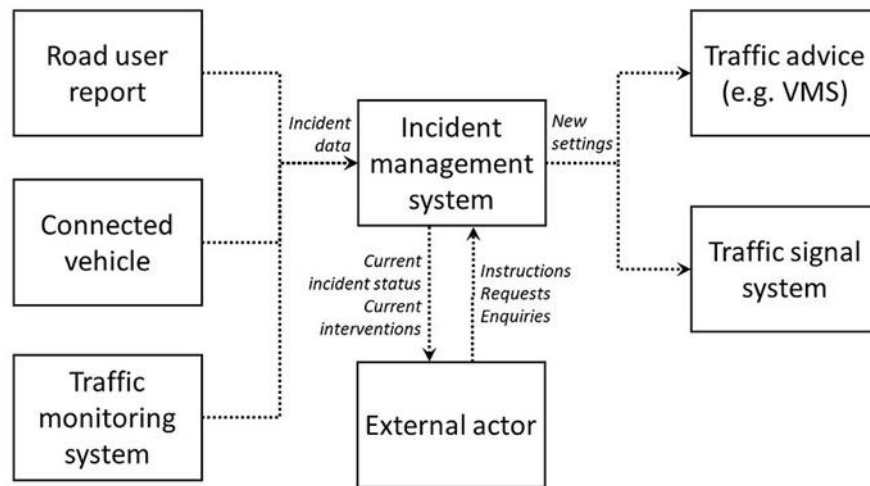


Рис. 1.6. Системи керування інцидентами[1]

Коригування фази сигналів для переміщення черг трафіку або (що більш ефективно) для керування новим рівнем попиту після впливу перерахованих вище інструментів.

Управління дорожніми роботами

Ця функціональна зона призначена для керування рухом у безпосередній близькості від дорожніх робіт.

У багатьох відношеннях використання ITS для підтримки цієї функціональної області віддзеркалює використання для управління інцидентами.

Управління якістю повітря

Метою цієї функціональної області є зміна, наскільки це можливо, розташування та поведінки учасників дорожнього руху, щоб забезпечити прийнятну якість повітря на всій території.

Ця діяльність включає деякі або всі з наступних аспектів:

Моніторинг поточного рівня якості повітря.

Моделювання можливої зміни якості повітря з урахуванням очікуваного розвитку дорожнього руху.

Надання інформації та, за необхідності, попереджень про очікувані рівні якості повітря.

Налаштування систем контролю для пом'якшення негативних наслідків, наприклад, шляхом переміщення черг із районів з (або очікуваною) низькою якістю повітря.

Зворотній зв'язок інформації та керуючих рішень у моделі. Приклад на рисунку 1.7.

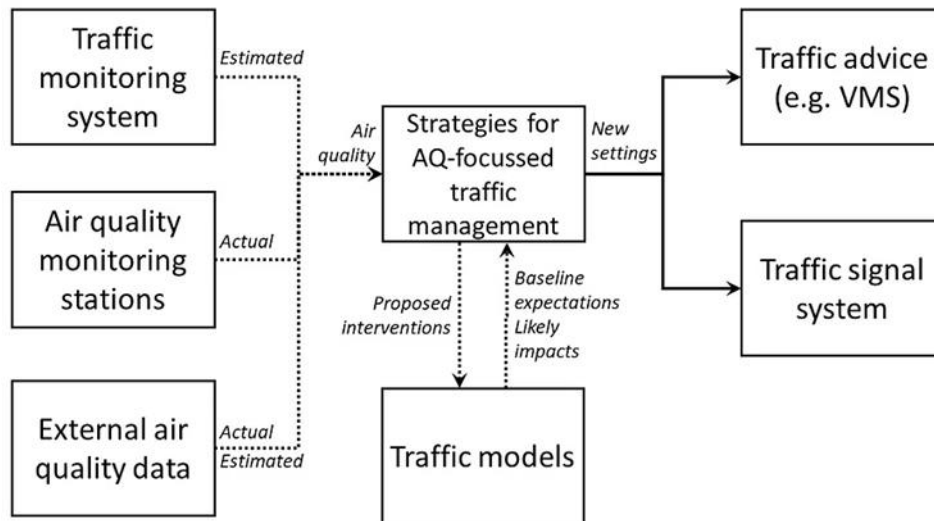


Рис.1.7. Системи керування якістю повітря[1]

Існує безліч варіантів того, як ці функції можуть бути включені до архітектури систем управління дорожнім рухом. На схемі нижче показано один із варіантів.

Інформація про дорожню мережу та поради

Метою цієї функціональної галузі є інформування нинішніх та потенційних учасників дорожнього руху про ситуацію на дорогах та надання їм рекомендацій щодо складання належних планів маршруту та часу їхньої поїздки.

Існує безліч видів тактичної інформації, яка може надаватися про робочий стан та поведінку дорожньої мережі. Це включає такі елементи : топологія мережі, обмеження швидкості, системи з одностороннім рухом, вузькі дороги, круті пагорби, місця інтересу, збори та мита, тощо.

Системи, що беруть участь у наданні інформації про дорожню мережу, бувають двох типів, що ґрунтуються на взаємодії залучених учасників: системи,

які використовуються диспетчером дорожнього руху для безпосереднього надання інформації, та системи, в яких вона надається через третю сторону.

Системи прямого надання включають встановлення та використання VMS, а також використання каналів CCAM для інформаційних послуг в автомобілі.

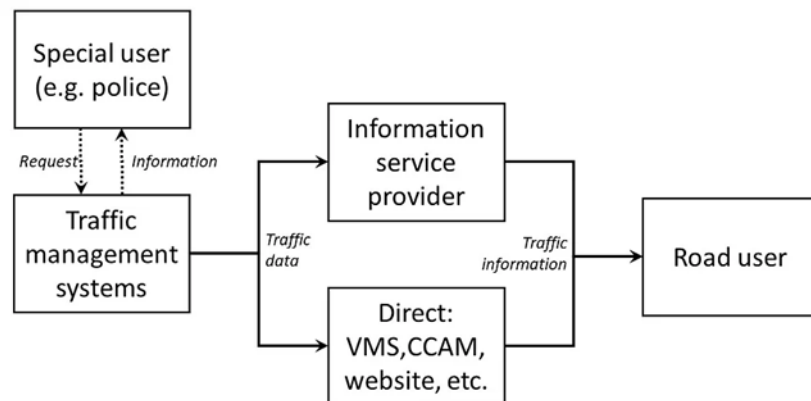


Рис.1.8. Системи інформації та рекомендацій щодо дорожньої мережі[1]

Непрямі системи фокусуються з точки зору диспетчера трафіку на публікації відповідного інформаційного потоку даних.

Оптимізація дорожньої мережі

Метою цієї функціональної області є розвиток дорожньої мережі таким чином, щоб максимізувати цілі дорожньої політики.

Зміни можуть бути потрібні у відповідь на виклики попиту (наприклад, нові райони будівництва або зміна землекористування) або політичні виклики (наприклад, скорочення викидів, заохочення активних подорожей).

Існує складна взаємодія з іншими аспектами управління транспортом, наприклад з організацією роботи громадського транспорту на місцевому рівні, а також з нетранспортними питаннями, такими як діяльність з економічного розвитку. У цьому поєднанні роль диспетчера дорожнього руху рідко полягає в тому, щоб керувати, а частіше у тому, щоб підтримувати ширші міські програми.

Оптимізація може включати фізичну реструктуризацію, наприклад, будівництво доріг або розширення доріг, зміни в правилах використання,

наприклад, обмеження на вантажні перевезення, зниження обмежень швидкості або зміни правозастосування, наприклад, введення камер контролю швидкості. Часто потрібне поєднання цих заходів, наприклад коли кільцева розв'язка замінюється регульованим перехрестям з пішохідними переходами і порушенням правил світлофора.

Зміни в мережі включатимуть три етапи:

- розуміння попиту використання доріг з погляду потреб відправлення та призначення, часу доби і типу транспортного засобу.
- розробка варіантів мережевої архітектури та перевірка того, як вони можуть вплинути на те, як учасники дорожнього руху обирають спосіб пересування.
- вибір структури цілей та розробка програми робіт для її реалізації з урахуванням таких факторів як вартість.

Спеціальне керування маршрутами користувачів

Метою цієї функціональної області є підтримка використання маршрутів для певних груп учасників дорожнього руху відповідно до цілей дорожньої політики. Приклад системи на рисунку 1.9.

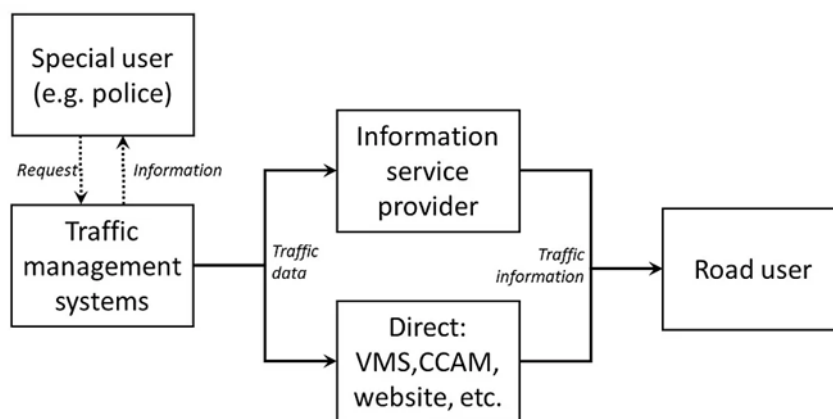


Рис.1.9. Система управління для спеціального транспорту[1]

Типові варіанти використання цієї функції включають:

- Автобусні маршрути за розкладом;
- Рекомендовані вантажні маршрути;

- Переважні маршрути обходу;
- Пріоритетні велосипедні маршрути;
- Зелені хвилі на пріоритетних дорогах, наприклад, радіальні маршрути в годину пік;
- Маршрути аварійно-рятувальних служб та маршрутизація.

Забезпечення дотримання правил дорожнього руху

Призначення цієї функціональної області – виявлення недотримання учасниками дорожнього руху правил дорожнього руху; виявляти невідповідних учасників дорожнього руху; і накладати, у разі потреби, відповідні покарання.

Багато правил дорожнього руху є обов'язковими для учасників дорожнього руху та, в принципі, відкриті для дотримання. Ступінь примусового виконання може змінюватись в залежності від обставин; місцева політика (і прагматика, така як доступне фінансування) може призвести до того, що це згодом зміниться, незалежно від змін самих правил.

Типові варіанти використання цієї функції включають:

- Перевищення швидкості;
- Проїзд на червоне світло (у тому числі на червоне світло на пішохідних переходах);
- Несанкціонований в'їзд (наприклад, на дорогу, заборонену для великовантажних автомобілів);
- Несанкціоноване використання проїжджої частини (наприклад, власний автомобіль на автобусній смузі);
- Проїзд хибним шляхом;
- Порушення викидів;
- Несанкціоноване паркування (у тому числі прострочене);

1.4. Основні компоненти для аналізу та моделювання

1.4.1. Система світлофорів

Світлофори є одним із найважливіших елементів системи дорожнього руху. Час очікування, коли світлофор стане зеленим, може становити значну частину часу в дорозі. Ефективна робота світлофорів може призвести до загальної ефективності руху.

Поширені проблеми включають необхідність чекати своєї черги на переїзд, коли з іншого боку не їде інший транспортний засіб, однаковий час очікування для руху на дорогах з високою і малою щільністю руху, фіксований час роботи світлофора протягом дня, непристосованість до змін тенденції дорожнього руху, необхідність чекати занадто багато змін світлофора під час подорожі дорогою з високою інтенсивністю руху тощо. Очевидно, що розумні світлофори, які працюють у інтелектуальній транспортній системі, здатні подолати ці проблеми.

Запропонована система світлофорів враховує як кількість транспортних засобів, так і час роботи. Передбачається, що кількість транспортних засобів, які очікують перетину на кожному переході, відома заздалегідь. У випадку підключених до мережі напів-автономних транспортних засобів це завдання є тривіальним, оскільки місцезнаходження транспортних засобів достатньо добре відоме глобальній системі позиціонування (GPS) або локальному алгоритму картографування. На рисунку 1.10 показано статистичне доведення переваги застосування розумних світлофорів.

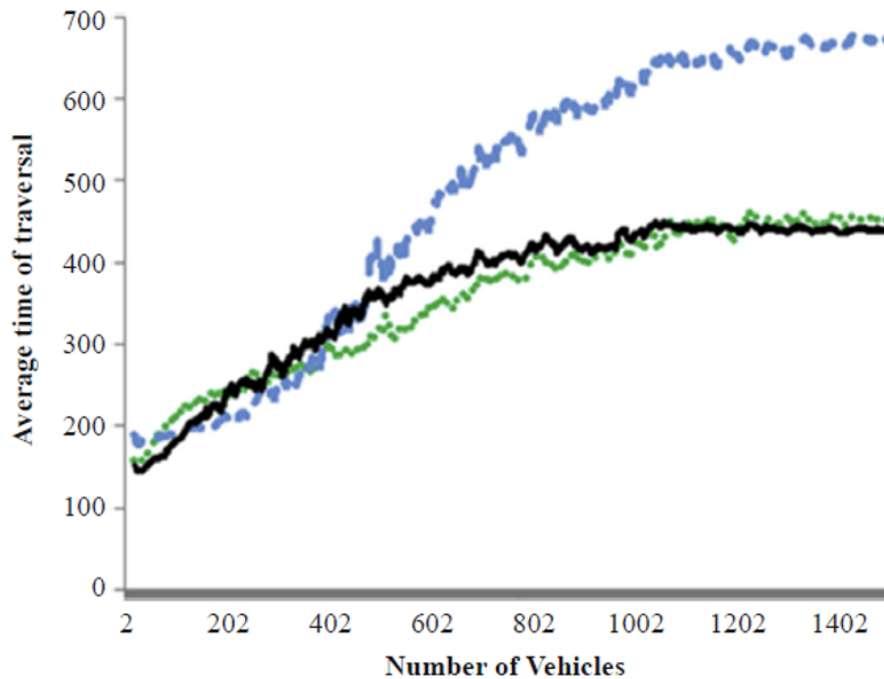


Рис. 1.10. Графік переваги системи керування світлофором [16]

Загальною тенденцією для всіх кривих було збільшення середнього часу проходження зі збільшенням кількості транспортних засобів, поки цей час не став постійним. Підвищення відбулося через збільшення інтенсивності руху. При досягненні перевантаженої інтенсивності руху будь-яке подальше збільшення кількості транспортних засобів не вплинуло. Крива з фіксованим часом роботи світлофора виявилася набагато неефективнішою порівняно з двома іншими підходами, які показали майже таку саму тенденцію. Тим не менш, для середньої щільності трафіку запропонована система справді перевищує систему з циклічними змінами перетину з точки зору середнього часу проходження.

1.4.2. Швидкісні смуги

Ідея призначення різних обмежень швидкості для різних смуг руху є важливою концепцією, яка особливо актуальна, якщо рух на дорозі відрізняється великою різноманітністю транспортних засобів, починаючи від транспортних засобів з низькою швидкістю до транспортних засобів з високою швидкістю. Очевидно, що рух повільніших транспортних засобів на всі смуги сповільнить рух на всіх смугах, що не підходить для транспортних засобів з високою

швидкістю. Подібно до випадку фіксованого часу роботи світлофора, оптимальність фіксованого обмеження швидкості для різних смуг руху є сумнівною. Таким чином, ми маємо динамічно регулювати обмеження швидкості на різних смугах руху.

1.4.3. Планування маршруту

Маршрутизація відіграє важливу роль у розподілі трафіку по мережі доріг, дозволяючи кожному транспортному засобу дістатися місця призначення за найкоротший час у взаємодії з іншими транспортними засобами. Насправді часто багато транспортних засобів їдуть популярною дорогою, що забезпечує швидкий доступ до певного пункту призначення. Однак це призводить до збільшення заторів і зниження швидкості руху для всіх, що призводить до зниження продуктивності. Неврахування інших транспортних засобів під час планування власного маршруту може призвести до поганих результатів. Рішення полягає в розумному розподілі трафіку на дорогах, використовуючи всю транспортну інфраструктуру для колективних подорожей. Альтернативна довша дорога може бути використана, якщо вона має меншу інтенсивність руху порівняно з основною дорогою. Однак, якщо альтернативна дорога занадто довга, вибір може бути невігідним.

У цьому підході для планування маршруту кожного транспортного засобу використовується алгоритм Uniform Cost Search. Метою алгоритму Uniform Cost Search є мінімізація часу подорожі всіх транспортних засобів. На будь-яку дорогу, обрану для руху, додається штраф, пропорційний інтенсивності руху на дорозі. Центральна інформаційна система знає поточну щільність руху на дорогах, але не знає очікувану щільність руху в очікуваний час прибуття транспортного засобу на дороги. Таким чином, щільність руху прогнозується з використанням історичної інформації про транспортний потік. Для доріг поблизу поточного місця розташування транспортного засобу поточна щільність є більш актуальною, оскільки вона не зміниться суттєво, доки транспортний засіб не прибуде на цю дорогу. Однак для доріг, розташованих далеко від поточного

положення транспортного засобу, передбачення на основі історичних даних є більш важливим, оскільки поточний сценарій руху може різко змінитися.

Маршрут, розрахований алгоритмом Uniform Cost Search, базується на поточному та прогнозованому трафіку, який змінюється з часом. Транспортні засоби можуть додати бажаний маршрут, транспортні засоби можуть звільнитися з бажаного маршруту, а нерегулярні тенденції можуть сильно відрізняти фактичний трафік від прогнозованого. Маршрут потрібно постійно адаптувати до цих мінливих тенденцій. Постійна адаптація всіх транспортних засобів до сценарію дорожнього руху дозволяє транспортним засобам спільно складати ефективний план подорожі. Ця адаптація здійснюється шляхом перепланування маршруту. Коли транспортний засіб опиняється на дорозі, вважається, що він не поверне назад, навіть якщо поворот назад веде до кращого маршруту. Отже, максимальна адаптація відповідає переплануванню при досягненні кожного перетину. Потім перепланований маршрут відображає будь-які змінені тенденції руху.

Формула 1.1[16]

$$\rho(R\alpha, t) = \begin{cases} \rho_{current}(R\alpha), & t \leq \beta \\ \rho_{historic}(R\alpha, t), & t > \beta \end{cases}$$

Тут $\rho_{current}(Ra)$ – поточна щільність трафіку, а $\rho_{historic}(Ra, t)$ – прогнозована щільність руху. β – час, до якого поточна щільність руху є надійною. Загальна сума - вартість, обчислена для вузла c_1 при розширенні з вузла c_2 , з'єднаного дорогою Ra для отже транспортний засіб V_i може бути задано рівняннями:

Формула 1.2[16]

$$t(c_1) = t(c_2) + \frac{|Ra|}{s_i}$$

$$f(c_1) = f(c_2) + a\rho(Ra, t(c_1))$$

На рисунку 1.11 показана динаміка системи в залежності від показника альфа.

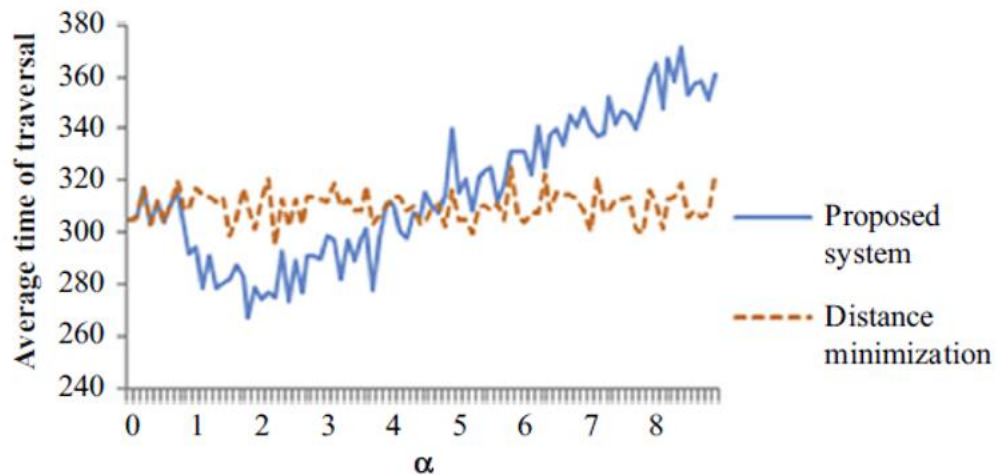


Рис.1.11. Результат застосування системи маршрутизації з змінною альфа[16]

Тут - альфа — постійна штрафу, $t(c1)$ — час прибуття до $c1$, Ra — довжина дороги.

1.4.4. Уникнення заторів у міському русі

Велика кількість транспортних засобів у межах дорожньої мережі зазвичай призводить до заторів, які відзначаються значним падінням середньої швидкості транспортних засобів, що рухаються. У результаті кожному транспортному засобу потрібен значний час, щоб дістатися до кінцевого пункту призначення. На конкретній дорозі затори можуть бути періодичними або одноразовими. Незважаючи на те, що звичайні водії зазвичай схильні коригувати час відправлення та маршрути для регулярного руху, одноразові затори важко передбачити та пристосуватися до них. Одноразові затори спричинені надзвичайно високим попитом (наприклад, спортивна подія) або раптовим зниженням пропускної здатності (як-от аварія чи перекриття дороги).

Зростаючий рівень автономності транспортних засобів і систем управління транспортом дав поштовх до можливостей уникнення заторів. Незважаючи на те,

що інтелектуальні агенти, пов'язані з дорожньою інфраструктурою, можуть визначати, відстежувати та вимірювати щільність руху на різних дорогах, інтелектуальні пристрої в транспортних засобах здатні збирати поточні дані та використовувати їх для цілей планування. Це дає змогу транспортному засобу об'їжджати дороги, на яких може виникнути затор, і використовувати альтернативні маршрути.

Заторів на дорозі можна значною мірою уникнути за допомогою методів маршрутизації, які вказують транспортному засобу маршрут, який їм потрібно проїхати. Встановлені на даний момент пристрої та системи, такі як супутникова навігація, приймають лише статичні дані. На жаль, це призводить до того, що одними дорогами їдуть кілька транспортних засобів, що призводить до заторів.

Маршрутизація може бути класифікована на централізовані підходи і децентралізовані підходи. Децентралізовані підходи розглядають кожен транспортний засіб окремо під час створення плану і, отже, можуть створити план подорожі за короткий час. Під час планування кожного транспортного засобу децентралізовані методи можуть віддавати перевагу (1) не враховувати рух іншого транспортного засобу, (2) прогнозувати рух інших транспортних засобів або (3) використовувати інформацію про прогнозування трафіку з історичних даних. Метод (1) призводить до великої завантаженості трафіку, а метод (3) не враховує одноразовий трафік. Мікро моделювання є звичайним інструментом для методу (2), у якому передбачається, що інформація про подорожі всіх транспортних засобів доступна, і система працює шляхом імітації різних можливих планів.

Цей метод має обмеження, включаючи той факт, що складно змоделювати велику кількість транспортних засобів для кожного перепланування траєкторії транспортного засобу або у випадку в'їзду будь-якого нового транспортного засобу. Якщо транспортний засіб переплановується, це може вплинути на плани деяких інших транспортних засобів, і іноді транспортним засобам може знадобитися багато часу, щоб отримати найкращі плани. Усі транспортні засоби на дорозі мають бути розумними, і зазвичай припускають, що вони мають

однакову швидкість. Крім того, невизначеності моделювання можуть стати дуже великими з часом. Ці невизначеності особливо великі при обліку обгону та сигналів світлофора. Все це робить акцент на те, що довгострокові плани є менш корисними для міського руху.

1.4.5. Децентралізована передбачувана маршрутизація

Зроблено значну спробу використати децентралізований випереджальний підхід до маршрутизації транспортних засобів. Була представлена система, в якій кожен транспортний засіб розглядає всі можливі маршрути, перш ніж вибрати маршрут для своєї подорожі. Автори реалізували формулу для перетворення очікуваної інтенсивності руху в середню швидкість руху. Це розширює роботу, яка використовувала мікро симуляцію руху для обчислення очікуваної швидкості руху. Перепланування виконується через певні часові кроки для кожного автомобіля. З огляду на обговорення, обмеженнями підходу є надто рідкісні перепланування, неможливість обчислення всіх можливих маршрутів у режимі реального часу для великих карт, припущення, що кожен транспортний засіб є розумним, відсутність уваги до світлофорів чи обгону і, нарешті, до всіх транспортних засобів.

1.5. Інструменти моделювання та симуляції архітектури ІТС

На основі вже встановленого набору стандартів ІТС було створено низку проектів, деякі з яких були орієнтовані на реалізацію/демонстрацію та моделювання ІТС. Серед них FRAME, HARTS, ARC-IT, SUMO.

FRAME (ФРЕЙМ)

Архітектура FRAME призначалася для використання в рамках низхідного підходу до планування та розгортання інтегрованих ІТС. Загальна концепція то, можливо і може бути представлена у формальній (еталонній) моделі. Оскільки створення еталонної моделі вимагає ухвалення низки рішень чи вибору тими, хто реалізує та/або регулює ІТС, архітектура FRAME їх не передбачає.

Загальна концепція та структура системи мають бути описані технологічно незалежним способом, щоб у міру розвитку технології вимоги вищого рівня залишалися незмінними. Відмінною рисою архітектури FRAME є те, що вона була розроблена з точки зору підмножин, тому навряд чи її можна буде використовувати повністю. На рисунку 1.12 показана архітектура фреймворку.

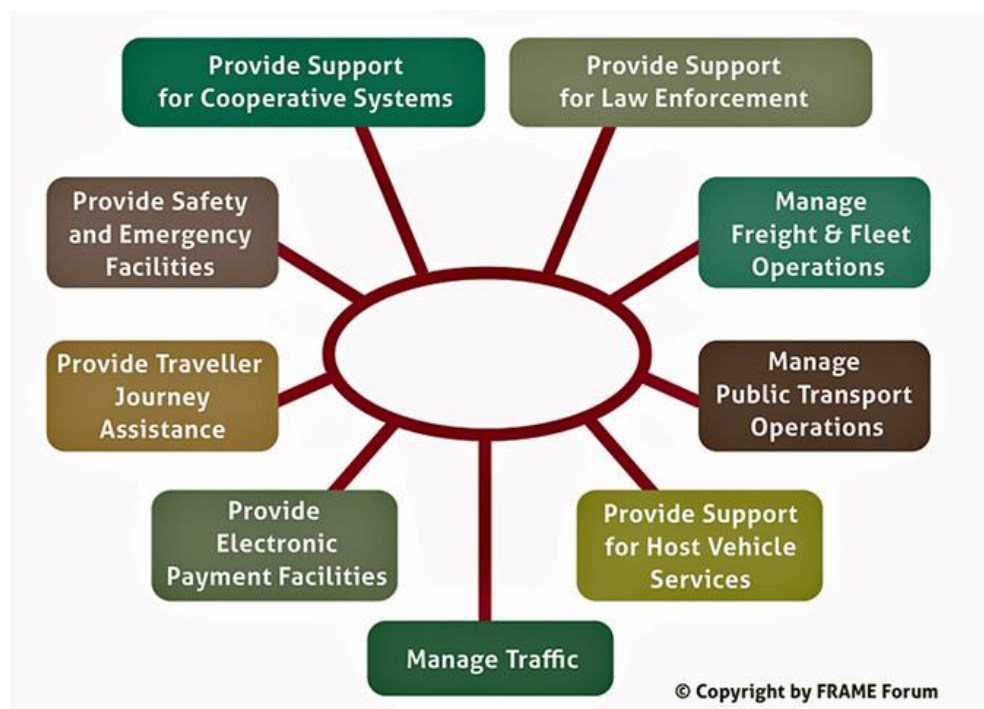


Рис.1.12. Архітектура інструменту моделювання FRAME [20]

Таким чином, архітектура FRAME є не так моделлю інтегрованої ІТС, як структурою, на основі якої можна систематичним і загальним чином створювати конкретні моделі інтегрованої ІТС.

HARTS (ХАРТС)

HARTS (Довідник з гармонізованої архітектури для технічних стандартів) був продуктом Цільової групи з гармонізації 7 (HTG7), яка була частиною серії Цільових груп з гармонізації, створених відповідно до угоди, укладеної у 2011 році між ЄС та США з метою підтримувати «глобальні відкриті стандарти скрізь, де це можливо, щоб забезпечити сумісність систем співробітництва в усьому світі та запобігти розробці та прийняттю надлишкових стандартів».

Одним з основних результатів HARTS є так звані пакети послуг ІТС, тобто набір орієнтованих на обслуговування перспектив систем ІТС.

Національна еталонна архітектура ITS США, також відома як «Довідник з архітектури для спільного та інтелектуального транспортування» або просто «ARC-IT», забезпечує загальну основу для планування, визначення та інтеграції інтелектуальних транспортних систем. ARC-IT є еталонною архітектурою, що забезпечує загальну основу для планувальників та інженерів з різними завданнями для планування, проектування та реалізації систем з використанням загальної мови як основи для надання ІТС, але не вимагає будь-якої конкретної реалізації. ARC-IT пропонує інструменти для планування та проектування систем ІТС.

ARC-IT

Архітектурний довідник для кооперативного та інтелектуального транспорту (ARC-IT) забезпечує загальну структуру для планування, визначення та інтеграції інтелектуальних транспортних систем. Продукт, який відображає внесок широкого кола ІТС-спільноти (спеціалісти з транспорту, системні інженери, розробники систем, спеціалісти з технологій, консультанти тощо). Приклад моделювання представлено в Додатку А,Б,В.

SUMO

«Simulation of Urban Mobility», або скорочено «SUMO», — це мікроскопічне мультимодальне моделювання руху з відкритим кодом. Це дозволяє змодельовати, як заданий транспортний попит, який складається з окремих транспортних засобів, рухається через дану дорожню мережу. Симуляція дозволяє розглянути великий набір питань управління трафіком. Він суто мікроскопічний: кожен транспортний засіб явно моделюється, має власний маршрут і рухається мережею окремо. Симуляції є детермінованими за замовчуванням, але існують різні варіанти введення випадковості. Більш детальний опис надано в 3 главі.

1.5.1. Основні компоненти моделювання

Процес моделювання транспортних потоків, має декілька ключових концепцій та характеристик на яких він базується. У таблиці 1.2 наведено приблизний перелік таких компонент.

Таблиця 1.2

Ключові компоненти керування транспортним потоком [16]

№	Концепція	Характеристики
1.	Мета маршрутизації	Щільність руху, контроль заторів, ризик, світлофори, очікуваний час у дорозі, найкращий/найгірший час у дорозі, час до місця призначення, час початку, зарезервована дорога (вартість подорожі)
2.	Частота маршрутизації	Часті перепланування, фіксовані плани, неповні або повні плани
3.	Припущення щодо маршрутизації трафіку	Повторювані, неповторювані, повторювані з деякою можливістю неповторюваних тенденцій
4.	Світлофор	Циклічний, фіксований, залежний від ситуації
5.	Зміна смуги руху	На основі обгону (додаткова смуга в основному використовується для обгону), спільна для транспортних засобів, які прибувають пізніше, на основі динамічного обмеження швидкості, зарезервована смуга (вартість проїзду)
6.	Трафік	Повністю пів автономний, змішаний, ручний

Цей перелік дозволяє при побудові моделей транспортного руху акцентуватись саме на такій структурі і будувати системи ІТС враховуючи ці компоненти.

1.5.2. Джерела інформації для системи ІТС

Крім компонент, безумовно, для вдалого моделювання транспортних потоків та розробки способів управління ними, необхідно мати достатній рівень інформаційного забезпечення, тобто даних, та проводити достатній аналіз для прийняття рішень. У таблиці 1.3 наведено основні джерела інформації, які можна використовувати для аналізу транспортних потоків та керування рухом.

Таблиця 1.3

Джерела формування первинних даних про рух [16]

Технології	Принципи	Переваги	Недоліки
Фреймворк адаптивної архітектури	Використання методів інтелектуального аналізу даних і алгоритмів машинного навчання для моніторингу трафіку	Жодних змін в існуючій системі не потрібно. Моніторинг автобусів і маршрутів в реальному часі	Поява нових джерел даних вплине на весь життєвий цикл даних і ефективність збору даних
Машинне навчання та великі дані	Транспортні засоби оснащені датчиками, які передають дані в реальному часі	Простий і економічний	Проблеми з точністю та надійністю, коли обсяги та щільність трафіку змінюються
Запис звуку (мікрофони)	Оцінка доплерівського зсуву автомобільних сигналів	Економічний, може використовуватися там, де рух не організований	Помилка в результатах зі збільшенням швидкості транспортного засобу, накладання гудків вплине на результати

Таблиця 1.3 (продовження)

Джерела формування первинних даних про рух [16]

Технології	Принципи	Переваги	Недоліки
GPS. акселерометр і мікрофон	Розроблений Nericell фактично переорієнтовує акселерометр на смартфоні, виявляє гудки та виконує локалізацію.	Енергоефективний, підходить для поганих доріг, хаотичного руху	Концентрація конфіденційності користувачів смартфонів, залучення користувачів до спільної системи
Бездротова мережа з трьома передавачами та приймачами	Інформація RSSI використовується для виявлення транспортних засобів і класифікації типу транспортного засобу.	Висока класифікація	Використані спрямовані антени роблять це дорогим.
Бездротова мережа за допомогою чотирьох передавачів і приймачів	Інформація RSSI використовується для виявлення транспортних засобів і класифікації типу транспортного засобу.	точність	Використовувані спрямовані антени роблять це дорогим.
П'єзоелектричні датчики прискорення	Алгоритми аналізують амплітуди та частотний діапазон коливань	Висока точність класифікації, низьке енергоспоживання, простий монтаж	Погодні умови можуть вплинути на ефективність

Таблиця 1.3 (продовження)

Джерела формування первинних даних про рух [16]

Технології	Принципи	Переваги	Недоліки
Сейсмометр і геофон	Розповсюдження Псевдо-Вігнера-Вілля	Економічність, висока точність	Порогове значення CFAR збільшується, коли наближається СРА, що спричиняє падіння вікна виявлення до завершення події
Тридцять сім мікрофонів	Ентропійний метод Реньї	Мінімум помилкових тривог, хороше виявлення	Висока вартість обладнання, обмеження точності

Висновки до розділу 1

Перший розділ присвячений дослідженню поняття інтелектуальної транспортної системи, архітектури та основних компонентів, розглянуто стан регулювання та стандартизації та перспективи розвитку сучасної транспортної системи з використанням інструментів великих даних та штучного інтелекту..

1. Проведено дослідження сучасної архітектури та вибрано окремі напрями для дослідження та реалізації окремих елементів системи управління транспортом.

2. Визначені основні складності створення системи управління транспортом.

3. Було розглянуто основні підходи до побудови ІТС.

4. Визначено перелік сервісів для створення системи управління та моделювання ІТС.

Спираючись на проведений аналіз та дослідження архітектури ІТС, способів моделювання та аналізу, необхідно зосередити увагу на розв'язання наступних завдань:

- Вибрати для подальшого аналізу та реалізації компонент прогнозування, моделювання трафіку та управління трафіком за допомогою системи світлофорів та прогнозування транспортних потоків;

- Вибрати спосіб аналізу ефективності прогнозування та визначитись з інструментами аналізу на базі статистичних методів та/або методів глибинного навчання;

- Розробити архітектуру способу управління на базі модельного підходу та використати симулятор моделі (або окремих модулів) для демонстрації запропонованої концепції за допомогою програми моделювання SUDO;

- оцінити ефективність запропонованого підходу, виділити переваги та недоліки, а також вказати на сфери можливого застосування запропонованого рішення.

РОЗДІЛ 2

МОДЕЛІ АНАЛІЗА ТА УПРАВЛІННЯ ДОРОЖНІМ ТРАФИКОМ З ДОПОМОГОЮ НЕЙРОМЕРЕЖ

2.1. Категоризація моделей транспортних потоків

Моделі транспортних потоків можна класифікувати за різними критеріями (рівень деталізації, операціоналізація, представлення процесів). Ми обговоримо декілька класів моделей транспортних потоків, порядок яких ґрунтується на класифікації рівня деталізації (тобто (суб-)мікроскопічний, мезоскопічний та макроскопічний підходи до моделювання).

2.1.1. Моделювання транспортних потоків

Рух на дорогах можна покращити за допомогою польових досліджень та польових експериментів із реальним транспортним потоком. Проте, крім наукової проблеми відтворення таких експериментів, вирішальну роль відіграють також вартість і безпека. Через складність системи транспортних потоків аналітичні підходи можуть дати бажаних результатів. Таким чином, моделі транспортних потоків (симуляції), призначені для характеристики поведінки складної системи транспортних потоків, стали важливим інструментом в аналізі транспортних потоків та експериментуванні.

Залежно від типу моделі область застосування цих моделей транспортних потоків дуже широка, наприклад:

Оцінка альтернативних методів (динамічного) керування дорожнім рухом.

Проектування та тестування нових транспортних засобів (наприклад, геометричні конструкції).

Моделі операційних потоків, які є підмодулем в інших інструментах (наприклад, управління та оптимізація трафіку на основі моделей, а також динамічне призначення трафіку).

Моделі трафіку класифікуються за такими ознаками:

- Шкала незалежних змінних (безперервна, дискретна, напівдискретна);

- Рівень деталізації (субмікроскопічний, мікроскопічний, мезоскопічний, макроскопічний);
- Подання процесів (детерміноване, стохастичне);
- Операціоналізація (аналітична, імітаційна);
- Масштаб застосування (мережі, ділянки, зв'язки та перетину).

2.1.2. Рівень опрацьованості деталей трафіку.

Моделі трафіку можна класифікувати відповідно до рівня деталізації, з яким вони представляють системи трафіку. Ця категоризація може бути введена в дію шляхом розгляду виділених об'єктів трафіку та рівня опису цих об'єктів у відповідних моделях потоків. Пропонуємо таку класифікацію:

1. Субмікроскопічні імітаційні моделі (високодетальний опис функціонування субодиниць транспортних засобів та взаємодії з навколишнім середовищем).
2. Мікроскопічні імітаційні моделі (високодетальний опис, в якому виділяються та відстежуються окремі об'єкти).
3. Мезоскопічні моделі (середня деталізація).
4. Макроскопічні моделі (низька деталізація).

Мікроскопічна імітаційна модель описує як просторово-часову поведінку сутностей системи (тобто транспортних засобів та водіїв), так і їхню взаємодію з високим рівнем деталізації (окремо). Наприклад, для кожного транспортного засобу в потоці зміна смуги руху описується як докладний ланцюг рішень водіїв.

Подібно до мікроскопічних імітаційних моделей, субмікроскопічні імітаційні моделі описують характеристики окремих транспортних засобів у транспортному потоці. Однак, крім докладного опису поведінки водіння, також докладно моделюється поведінка керування транспортним засобом відповідно до навколишніх умов. У цьому описується функціонування конкретних елементів (вузлів) автомобіля.

Мезоскопічна модель не розрізняє та не відстежує окремі транспортні засоби, а визначає поведінку людей, наприклад, у ймовірнісних термінах. З цією метою трафік представлений (невеликими) групами об'єктів трафіку, дії та взаємодії яких описуються на низькому рівні деталізації. Маневр зі зміни смуги руху може бути представлений для окремого транспортного засобу як миттєва подія, коли рішення про зміну смуги руху засноване, наприклад, відносна щільність смуг руху і різниця швидкостей.

Макроскопічні моделі потоків описують трафік на рівні агрегації як потік без виділення його складових частин. Наприклад, потік трафіку представлений в агрегованому вигляді з використанням таких характеристик, як швидкість потоку, щільність та швидкість. Маневри окремих транспортних засобів, такі як зміна смуги руху, зазвичай не представлені. Макроскопічна модель може припускати, що транспортний потік правильно розподіляється смугами проїжджої частини, і для цього використовувати апроксимацію. Макроскопічні моделі течії можна класифікувати за кількістю диференціальних рівнянь у приватних похідних, які часто лежать в основі моделі, з одного боку, та їх порядку з іншого боку. Безперервна модель описує, як стан системи дорожнього руху безперервно змінюється з часом у відповідь на безперервні дії. Дискретні моделі припускають, що зміни стану відбуваються уривчасто в часі в дискретні моменти часу. Крім часу інші незалежні змінні можуть бути описані як безперервними, так і дискретними змінними (наприклад, положення, швидкість, бажана швидкість).

Подання процесів. У цьому відношенні ми будемо розрізняти детерміновані та стохастичні моделі. У перших моделях немає випадкових змінних, що означає, що всі дійові особи моделі визначаються точними відносинами. Стохастичні моделі включають процеси, що включають довільні змінні.

Операціоналізація.

Що стосується критерію операціоналізації, моделі можуть бути операціоналізовані або як аналітичні рішення систем рівнянь або як імітаційна модель.

Масштаб застосування. Шкала застосування вказує на сферу застосування моделі. Модель може описувати динаміку своїх об'єктів для окремої ділянки дороги, всієї транспортної мережі, коридору, міста тощо.

Що стосується застосування моделей, мікроскопічні моделі ідеально підходять для автономного моделювання, наприклад, для перевірки геометрії проїжджої частини. З точки зору застосування до заснованих на моделях оцінювання, прогнозування та управління відсутність закритого аналітичного рішення представляє проблему, яку нелегко вирішити.

Макроскопічні моделі підходять для великомасштабних мережних додатків, де основний інтерес представляють макроскопічні характеристики потоку. Як правило, калібрування макроскопічних моделей відносно просте (порівняно з мікроскопічними та мезоскопічними моделями). Однак макроскопічні моделі, як правило, занадто грубі, щоб правильно описати мікроскопічні деталі та дії, наприклад, викликані змінами геометрії проїжджої частини. Однак, через наявність закритих аналітичних рішень вони дуже підходять для застосування в оцінці, прогнозуванні та управлінні транспортним потоком на основі моделей.

2.1.3. Прогнозування трафіку

Прогнозування трафіку означає прогнозування обсягу та щільності транспортного потоку, зазвичай з метою управління рухом транспортних засобів, зменшення заторів та створення оптимального (з найменшими витратами часу чи енергії) маршруту.

Прогнозування трафіку переважно важливо для двох груп організацій (ми не говоримо про людей, які планують поїздку на вихідні).

Національні/місцеві органи влади. За останні десять-двадцять років багато міст запровадили інтелектуальні транспортні системи (ІТС), які підтримують

планування міської транспортної мережі та управління дорожнім рухом. Ці системи використовують поточну інформацію про дорожній рух, а також згенеровані прогнози для підвищення ефективності та безпеки транспорту, інформуючи користувачів про поточні дорожні умови та коригуючи дорожню інфраструктуру (наприклад, вуличне освітлення).

Логістичні компанії. Ще одним напрямом застосування є логістична галузь. Транспорт, доставка, виїзне обслуговування та інші підприємства повинні точно планувати свої операції та створювати найефективніші маршрути. Часто це пов'язано не лише з поточними поїздками, а й із заходами у майбутньому. Точні прогнози дорожніх умов та умов руху, щоб уникнути заторів, мають вирішальне значення для планування та роботи таких компаній.

На рисунку 2.1 схематично представлено перелік напрямів використання прогнозування транспортних потоків.

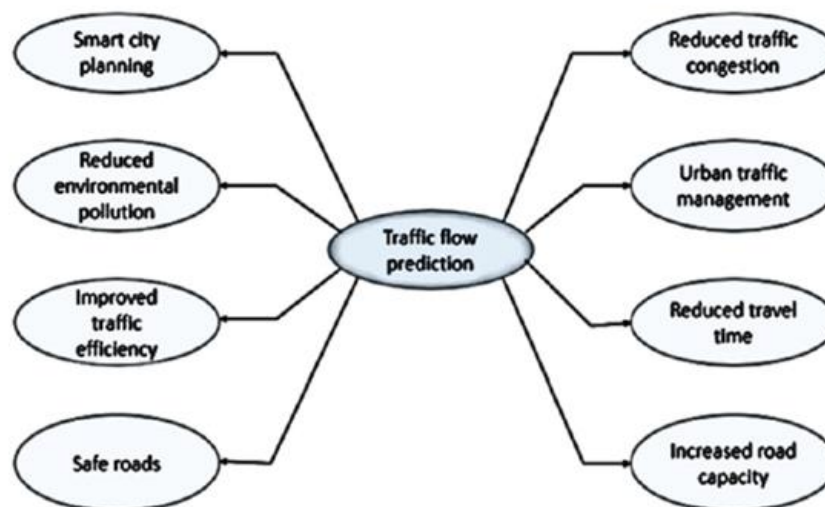


Рис.2.1. Переваги системи передбачення транспортних потоків[1]

Різні методи машинного навчання (і особливо глибокого навчання), здатні обробляти величезні обсяги як історичних даних, так і даних у реальному часі, використовуються для прогнозування потоку, щільності та швидкості трафіку.

2.1.4. Прогнозування заторів

Прогнозування транспортних заторів складається з двох основних етапів збору даних і розробки моделі прогнозування. Кожен крок методології важливий і може вплинути на результати, якщо його виконати неправильно.

На рисунку 2.2. показано загальні компоненти досліджень прогнозування заторів.

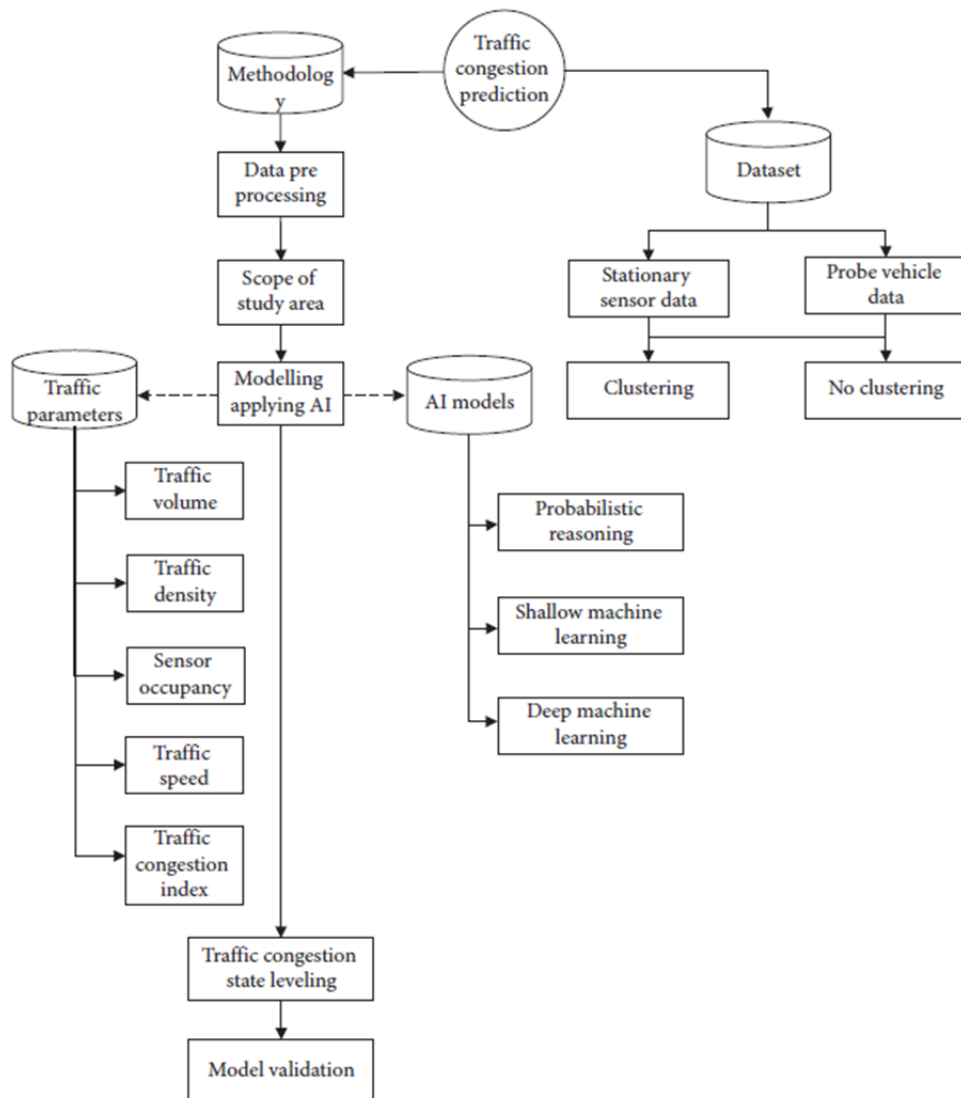


Рис. 2.2. Загальна система прогнозування заторів[12]

Після збору даних обробка даних відіграє важливу роль для підготовки наборів даних для навчання та тестування. Область справи різна для різних досліджень. Після розробки моделі її перевіряють іншими базовими моделями та обґрунтовують справжні результати.

2.1.5. Типи даних та джерела

Набори даних про рух, які використовуються в різних дослідженнях, можна в основному розділити на два класи, включаючи стаціонарні дані та дані зондів. Стаціонарні дані можна далі розділити на дані датчиків і дані фіксованих камер. З іншого боку, дані зонда, які використовувалися в дослідженнях, були даними GPS, встановленими на автомобілях. Стаціонарні датчики постійно фіксують просторово-часові дані руху. Однак робота датчика може перерватися в будь-який час. Органи влади повинні завжди враховувати цей тимчасовий збій датчика під час планування, використовуючи ці дані. Перевага даних датчиків полягає в тому, що немає плутанини щодо місця розташування транспортних засобів.



Рис.2.3. Дані для прогнозування трафіку[9,10]

Насамперед, вам потрібна докладна карта з дорожніми мережами та відповідними атрибутами. Підключення до таких глобальних постачальників картографічних даних, як Google Maps, TomTom, HERE або OSM, — чудовий спосіб отримати повну та актуальну інформацію.

Інформація про трафік, рисунок 2.4. Потім вам потрібно буде зібрати як історичну, так і поточну інформацію про дорожній рух, таку як кількість транспортних засобів, що проїжджають у певній точці, їх швидкість та тип

(вантажівки, легкові автомобілі тощо). Пристрої, що використовуються для збору цих даних,

- петлеві детектори,
- камери,
- вагові датчики руху та
- радары чи інші сенсорні технології.

Цю інформацію простіше отримати від вищезазначених постачальників, які збирають дані із системи датчиків, різних сторонніх джерел або використовують дані GPS-зондів.

Інші платформи, такі як Otonomo, використовують інноваційну технологію Vehicle to Everything (V2X) для збору даних про так звані підключені автомобілі з вбудованих модемів.

Ви також можете отримати іншу важливу інформацію про події (перекриття доріг або дорожні роботи), пам'ятки тощо від постачальників даних.

Інформація про погоду. Дані про погоду (історичні, поточні та прогнозовані) також необхідні, оскільки метеорологічні умови впливають на дорожню ситуацію та швидкість руху. Існує безліч постачальників даних про погоду, до яких можна підключитися, наприклад OpenWeather або Tomorrow.io.

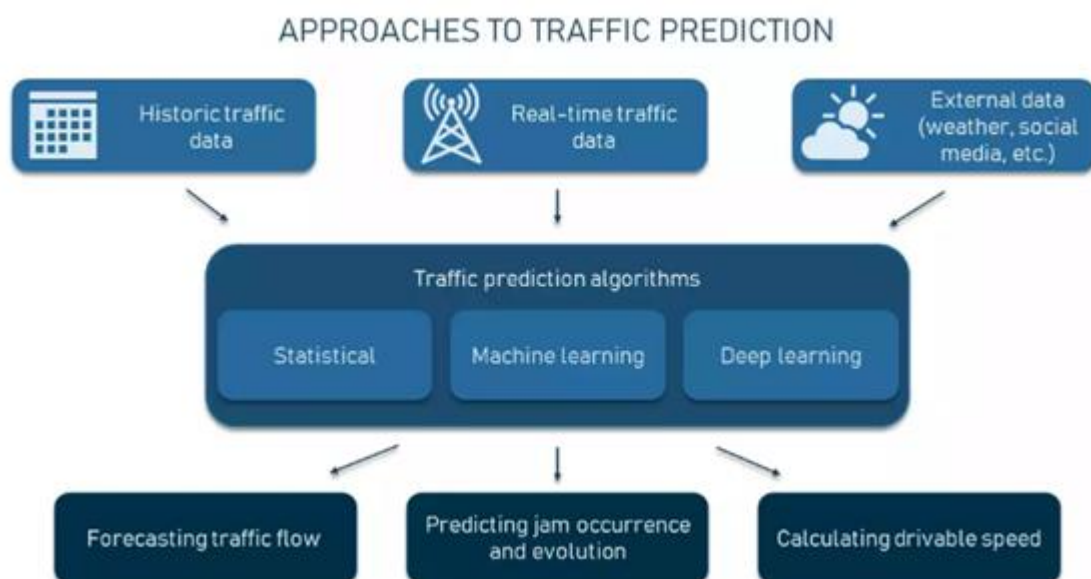


Рис.2.4. Підходи до прогнозування трафіку[9,10]

Додаткові дані щодо дорожніх умов. Існують зовнішні джерела даних, які можуть надати важливу інформацію, що впливає трафік. Подумайте про

повідомлення в соціальних мережах про спортивні події в цьому районі, місцеві новини про громадянські протести або навіть поліцейські сканери про місця злочинів, аварії або перекриття доріг.

2.2. Алгоритми аналізу транспортних потоків

2.2.1. Алгоритми кластеризації

Кластеризацію отриманих даних використовують перед застосуванням основних моделей прогнозування перевантаження. Ця техніка гібридного моделювання використовується для точного налаштування вхідних значень і їх використання на етапі навчання.

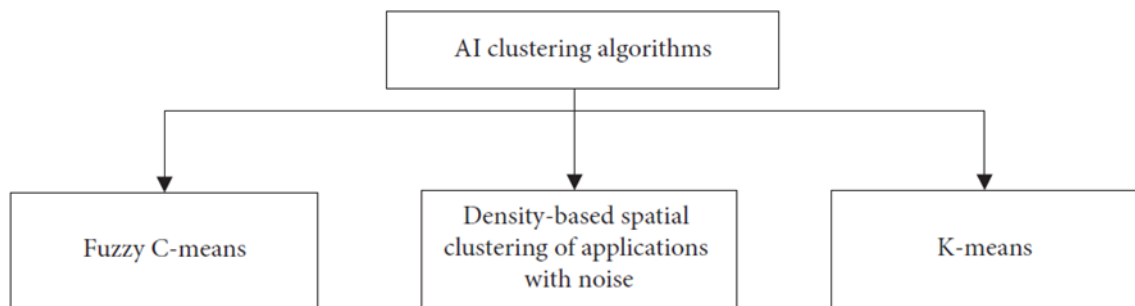


Рис 2.5. Алгоритми кластеризації[12]

На рисунку 2.5 показано моделі кластеризації III, які зазвичай використовуються в цій галузі досліджень. У цьому розділі коротко описано моделі.

2.2.2. Статистичний підхід

Статистичні методи дозволяють виявити закономірності трафіку в різному масштабі: протягом дня, у різні дні тижня, сезонні тощо. Однак вони менш точні, оскільки не можуть опрацьовувати стільки багатовимірних даних.

Зокрема, моделі авторегресійного інтегрованого ковзного середнього (ARIMA) активно використовувалися для прогнозування трафіку з 1970-х років, оскільки вони прості в реалізації та демонструють більш високу точність порівняно з іншими статистичними методами. Це класичний статистичний

підхід до аналізу минулих подій та прогнозування майбутніх. Він спостерігає за даними, які збираються із серії регулярних інтервалів часу, і припускає, що минулі моделі повторюватимуться у майбутньому.

Однак транспортний потік є складною структурою з багатьма змінними, які не можуть бути ефективно оброблені за допомогою одновимірних моделей ARIMA.

2.2.3. Підхід до машинного навчання

Машинне навчання (ML) дозволяє створювати прогностичні моделі, що враховують великі маси різноманітних даних із різних джерел. Було проведено багато досліджень щодо застосування алгоритмів машинного навчання для прогнозування дорожнього руху. На рисунку 2.6 представлена узагальнене дерево алгоритмів машинного навчання для управління транспортним потоком.

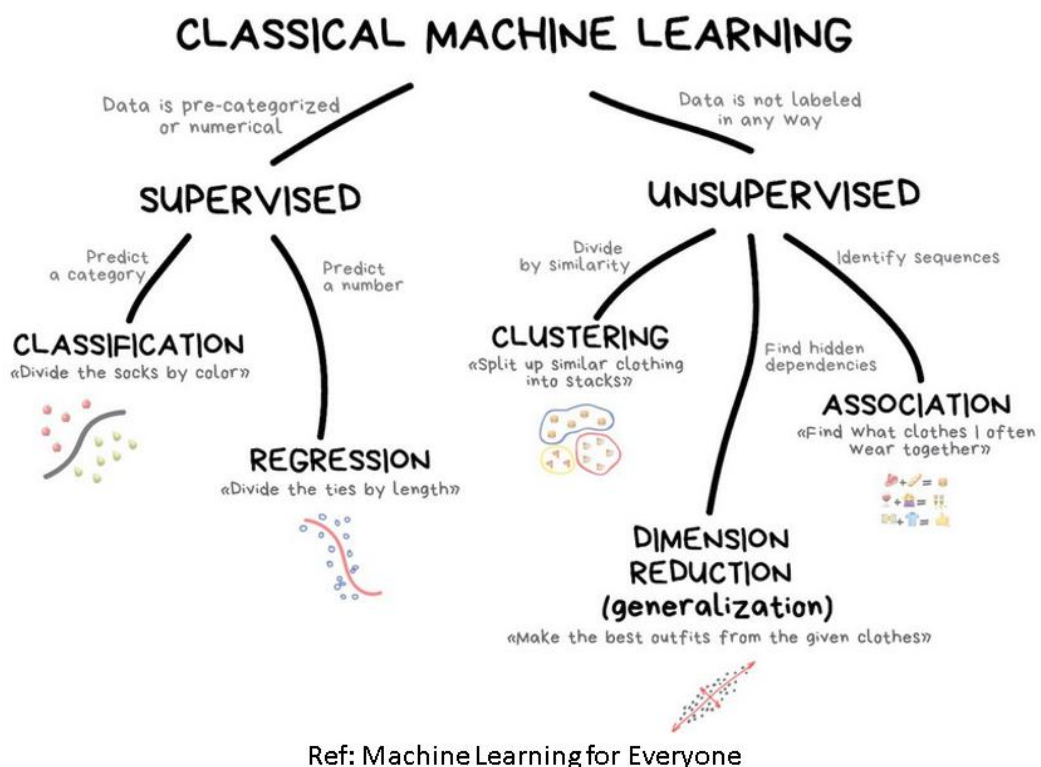


Рис 2.6 Дерево алгоритмів машинного навчання

Алгоритм випадкового лісу (Random forest) створює кілька дерев рішень та поєднує їх дані для отримання точних прогнозів. Це досить швидко і може давати ефективні результати за наявності достатньої кількості навчальних даних.

Щодо проблеми пробок на дорогах цей метод показав точність 87,5%. В цьому випадку як входні змінні моделі використовуються погодні умови, період часу, особливі умови дороги, якість дороги та свята.

Алгоритм k-найближчих сусідів (KNN) ґрунтується на принципі подібності ознак для прогнозування майбутніх значень. Експерименти з моделлю KNN продемонстрували більш ніж 90% точність короткострокового прогнозування транспортних потоків.

2.2.4. Підхід до глибокого навчання

Методи глибокого навчання (DL) виявилися дуже ефективними у прогнозуванні дорожнього руху порівняно з (ML) чи статистичними методами, постійно демонструючи точність прогнозування близько 90% та вище. Алгоритми глибокого навчання ґрунтуються на нейронних мережах.

Нейронні мережі (NN) або штучні нейронні мережі (ANN) складаються з взаємопов'язаних вузлів (нейронів), які розташовані у два або більше шарів і призначені для роботи аналогічно до людського мозку. Існує безліч типів нейронних мереж, розроблених для різних цілей. Ось деякі з них, які використовувалися при аналізі та прогнозуванні трафіку.

Згорткові нейронні мережі (CNN) є визнаними лідерами у розпізнаванні та аналізі зображень. Одним із їх природних застосувань до транспортних проблем є виявлення пробок з використанням зображень із камер спостереження на дорозі. Середня точність класифікації у разі досягає 89,5 відсотка. Що стосується прогнозування трафіку, то CNN — не найкращий вибір. Однак були досить успішні спроби побудови моделей на основі CNN, які прогнозують швидкість транспортної мережі. Щоб це сталося, дослідники перетворили тимчасові та просторові дані, що описують транспортний потік, на двовірну матрицю зображень.

Рекурентні нейронні мережі (RNN), на відміну CNN, призначені для обробки даних тимчасових рядів або спостережень, зібраних за певні інтервали часу. Моделі дорожнього руху є добрим прикладом таких спостережень.

Дослідження показали високу точність прогнозування розвитку навантажень при використанні моделей RNN. Однак їх недоліком є проблема градієнта, що зникає, що означає, що частина даних з попередніх шарів втрачається (ось чому кажуть, що RNN «мають короточасну пам'ять»). Ця «забудькуватість» робить навчання моделі більш важким і трудомістким.

Довготривала короточасна пам'ять (LSTM) і вентильована рекурентна одиниця (GRU) - це різновиди RNN, що вирішують проблему градієнта, що зникає. Дослідження, в якому порівнювалася продуктивність цих моделей, показало, що модель GRU точніше прогнозує транспортні потоки та її легко навчати. На рисунку 2.7 представлено дерево моделей.

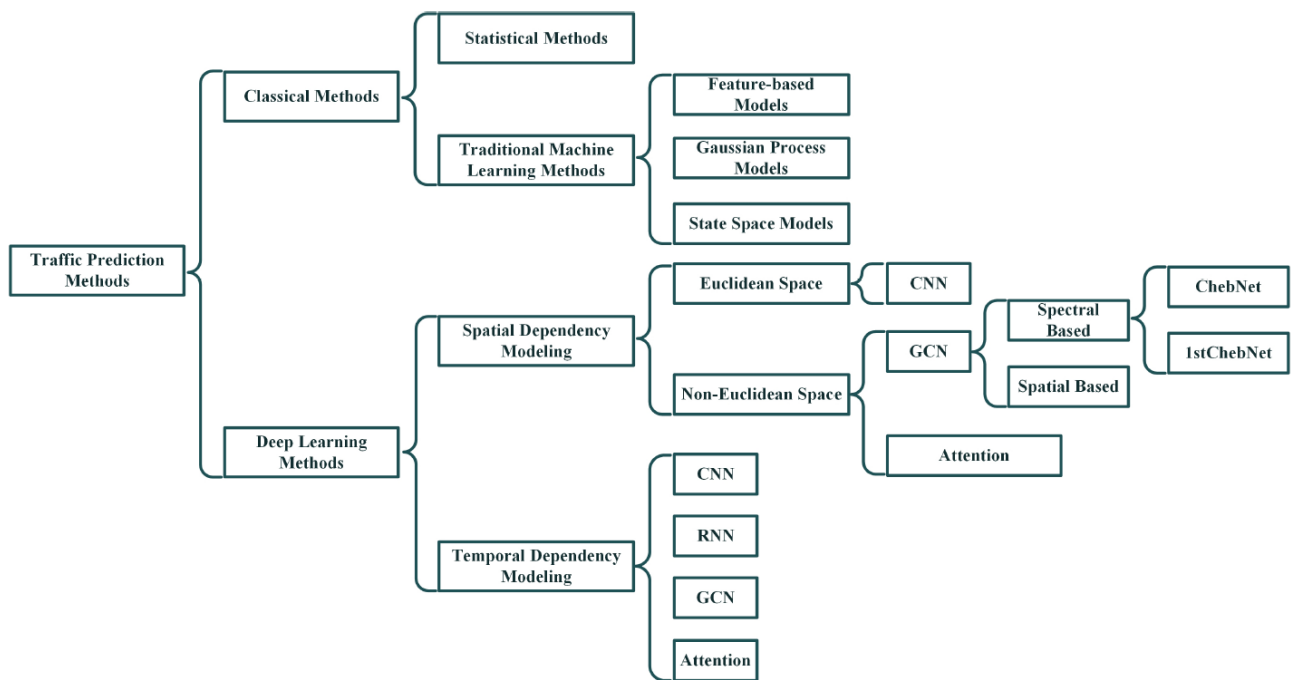


Рис.2.7. Ключові техніки прогнозування транспортних потоків[17]

Існує велика кількість досліджень, в яких пропонується будувати інші типи NN-моделей для прогнозування трафіку, наприклад, графові нейронні мережі, нечіткі нейронні мережі, байєсівські нейронні мережі та інші, а також використовувати гібридні методи, що поєднують два або більше алгоритмів. На сьогоднішній день не знайдено жодної найкращої методики, яка могла б застосовуватися у всіх випадках та давати найточніші прогнози.

2.2.5. Як реалізувати прогнозування трафіку

Якщо ви керуєте логістичним бізнесом, швидше за все вам потрібен не прогноз трафіку сам по собі, а його вплив на ваші операції. Як ми вже згадували, точний прогноз важливий для цілей маршрутизації та планування. Якщо це так, є три основні способи отримати ці прогнози та побудувати оптимальні маршрути.

2.2.6. Готові рішення

На ринку представлено багато готових програмних рішень, розроблених для будь-якого виду бізнесу. На рисунку 2.8 представлений певний перелік компаній.

MAIN TRAFFIC PREDICTION DATA PROVIDERS			
	Traffic prediction functionality	Frequency of updates	Routing functionality
Google Maps Platform	Only current traffic	Frequently	✓
Waze	Only current traffic	Every 2 minutes	✓
TomTom	Up to 24 hours ahead	Every 30 seconds	✓
HERE	Up to 12 hours ahead	Every minute	✓
ArcGIS	Up to 4 hours ahead	Every 5 minutes	✓
PTV	Up to 1 hour ahead	N/A	✓

Рис.2.8. Провайдери даних щодо прогнозування трафіку[10]

Платформи Google Карт. Якщо ви хочете співпрацювати з найбільшим постачальником картографічних даних, майте на увазі: ви зможете додати на карту лише поточний шар трафіку, але не прогнози.

Waze. Ще один продукт Google і другий за популярністю навігаційний додаток отримує інформацію про пробки, аварії, пробки та інші дорожні умови тільки від водіїв. Інформація оновлюється кожні 2 хвилини, і до неї можна

отримати доступ через локалізовані потоки даних XML та JSON Geo RSS. Ви також можете вбудувати Waze Live Map із функціями пошуку та маршрутизації на свій веб-сайт. Щодо прогнозів, то додаток їх не робить і не надає.

ТомТом. API TomTom Traffic RESTful надають доступ до історичних даних та даних у реальному часі, пов'язаних з дорожніми пригодами та потоками. TomTom використовує понад 600 мільйонів датчиків даних GPS та плаваючих автомобілів для збирання актуальної інформації (оновлюється кожні 30 секунд), її аналізу та прогнозування на 24 години вперед. Ці дані можна використовувати для створення маршрутів за допомогою власної програми або отримати комплексний продукт для маршрутизації.

HERE. Платформа HERE Real-Time Traffic збирає дані з понад 100 служб звітності про інциденти і щодня використовує мільярди точок даних GPS. Його дані оновлюються щохвилини. HERE надає інформацію в режимі реального часу та створює точні прогнози на наступні 12 годин на основі історичних та поточних даних про трафік. Подробиці дивіться у їх документації щодо Traffic RESTful API.

АркГІС. API-інтерфейси REST сервісу трафіку ArcGIS дозволяють візуалізувати умови трафіку у вашому додатку. Дані про трафік оновлюються кожні 5 хвилин, і доступні прогнози на наступні 4 години. Крім того, на веб-карті представлена інформація про події. PTV.

2.2.7. Використання нелінійних методів аналізу та прогнозування трафіку

Прогнозування заторів на дорогах значною мірою залежить від датчиків та іншого обладнання для збору відповідних даних, таких як швидкість руху, дані про погоду та аварії тощо. і ці традиційні моделі машинного навчання не працюють у поточних сценаріях. Тому давайте докладно розглянемо моделі глибокого навчання, а також досліджуємо область неконтрольованого навчання щодо прогнозування транспортних потоків.

У традиційних моделях використовувалися параметричні методи. Модель авторегресивного інтегрованого ковзного середнього (ARIMA) є добре відомою

структурою та еталоном для короткострокового прогнозування транспортних потоків. У модель ARIMA було внесено безліч змін, і результати показали покращення показників. Однак через стохастичний та нелінійний характер транспортного потоку параметричні моделі не дали точних результатів. [18]

Тому перевага надавалася непараметричним моделям, і тому нейронні мережі стали популярними для прогнозування транспортних потоків. Неглибока нейронна мережа зворотного поширення (BPNN) показала перспективні результати, але вона не працювала в епоху великих даних. Таким чином, з'явилося глибоке навчання, яке використовує кілька рівнів для поступового отримання ознак вищого рівня з необроблених вхідних даних. Багато архітектур глибокого навчання включають згорткову нейронну мережу (CNN), рекурентну нейронну мережу (RNN), довготривалу короткочасну пам'ять (LSTM), Обмежені машини Больцмана (RBM), Мережа глибокої довіри (DBN) та Stacked Auto Encoder (SAE).

Моделі глибокого навчання показали перспективні результати, що відображають нелінійність прогнозування транспортних потоків. Наприклад, було доведено, що популярні архітектури глибокого навчання, такі як CNN, адаптуються до просторових залежностей трафіку. RNN, особливо архітектури LSTM, пристосовуються до довгострокових та короткострокових тимчасових залежностей потоку трафіку.

Хоча використання згаданих моделей глибокого навчання для індивідуального прогнозування потоку трафіку має кілька переваг, вони мають і суттєві недоліки. Відповідно, оскільки сучасні транспортні системи покладаються на точні дані, будь-які відсутні дані серйозно впливають на точність моделі глибокого навчання і можуть призвести до неоптимальної чи неефективної роботи.

Ще одне обмеження архітектуро глибокого навчання, таких як CNN та RNN, полягає в тому, що для навчання їм потрібні великі обсяги історичних даних. Тут виконати вимогу досить просто, оскільки ми живемо в епоху великих даних. Однак, згідно з, великі обсяги історичних даних можуть призвести до

перепідгонки моделі через сильні коливання транспортного потоку на невеликому часовому інтервалі.

Таким чином, останнім часом дослідники починають переходити від архітектур глибокого навчання до гібридних та неконтрольованих методів. Нещодавні оглядові документи в основному були зосереджені на огляді моделей на основі машинного навчання, статистичних моделей та моделей прогнозування міського потоку для прогнозування дорожнього руху.

2.2.8. Глибокі нейронні мережі (DNN)

Оскільки на транспортний потік впливає безліч факторів, таких як погода, аварії, свята тощо, глибока архітектура підходить для вирішення складного завдання прогнозування транспортного потоку. Тепер це можливо після того, як Хінтон запропонував революційну глибоку мережу, яка перетворювала дані високої розмірності на дані низької розмірності, які перевершили PCA.

Автори (Tangubolon & Hsiung, 2018) продемонстрували методи налаштування гіперпараметрів для оптимізації мереж та отримання кращих результатів у задачі узагальнення та перенавчання. Вони використовували алгоритм зворотного розповсюдження для розрахунку градієнта та оновлення ваг за допомогою градієнтного стохастичного спуску (SGD). Був доданий шар, що відсівається, і виконана пакетна нормалізація, щоб забезпечити стандартизацію вхідного розподілу перед передачею активацій нелінійним шарам.

Каскадна штучна нейронна мережа (CANN) була запропонована, яка складається з трьох штучних нейронних мереж (ANN): довгострокової, середньострокової та короткострокової. Довгострокова мережа фіксує щотижневі повторення у потоці трафіку, середньострокова мережа фіксує щоденну періодичність у потоці трафіку, а короткострокова мережа фіксує числові зміни тенденцій потоку трафіку.. На рисунку 2.9 наведено основні нейронні мережі, які використовують для аналізу даних що стосуються транспорту.

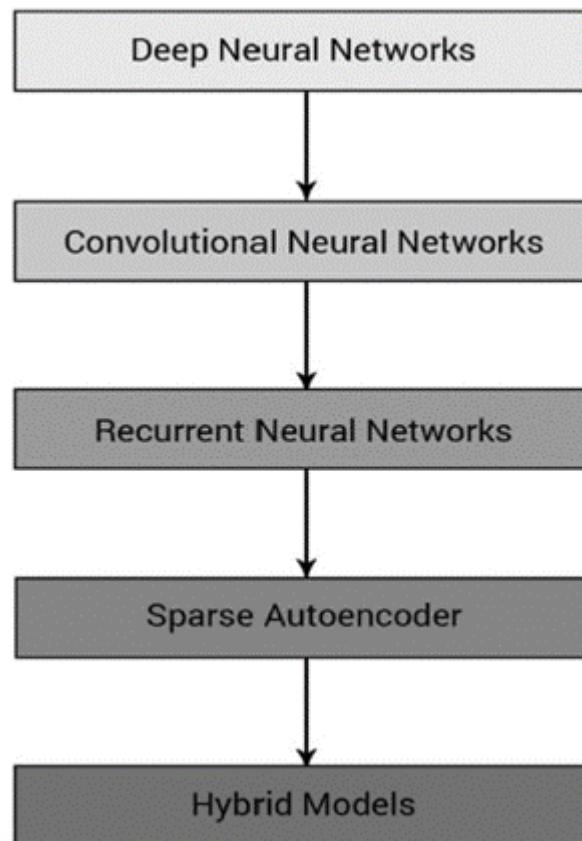


Рис.2.9. Архітектура моделей нейронних мереж[19]

Модель CANN показала найкращі результати під час використання історичних даних

2.2.9. CNN (згорточні нейронні мережі)

Це алгоритми глибокого навчання, які мають можливості приймати багатовимірні вхідні дані та виконувати складні обчислення з відносною легкістю порівняно зі звичайною DNN. Conv-мережі виявилися ефективним інструментом для захоплення просторово-часових характеристик за допомогою фільтрів, що самонавчаються. Шари об'єднання в Conv-Net підвищують здатність моделі до узагальнення та скорочують час обчислень. Ці особливості CNN є мотивацією для їх застосування в галузі прогнозування транспортних потоків.

2.2.10. SAE мережі (автоенкодери)

Одне з перших визначень автокодувальників було надано Баллардом у 1987 році, на рисунку 2.10 представлено приклад архітектури моделі.

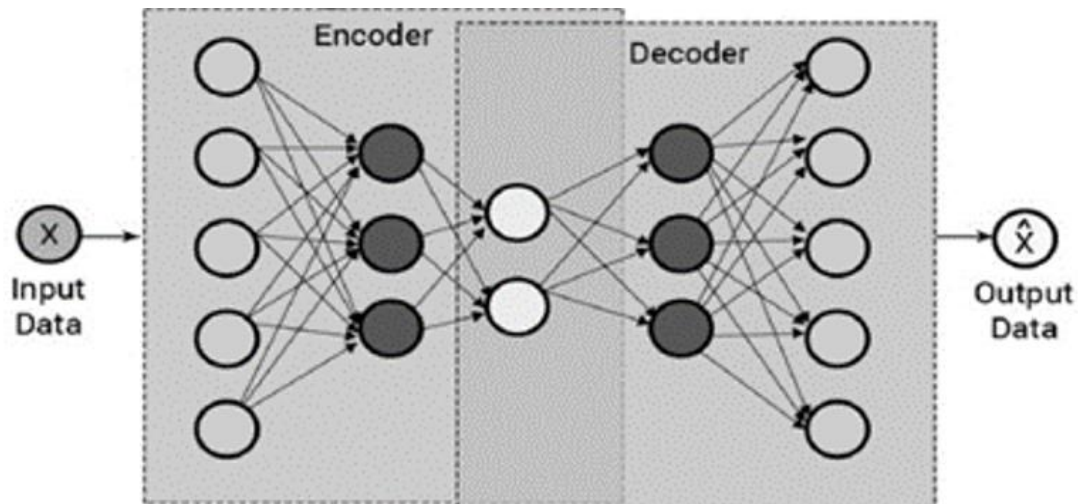


Рис.2.10. Базова архітектура автоенкодера[19]

Вони визначають автокодувальники або автоасоціативну конфігурацію як модель, в якій вихідні дані мають бути ідентичними для вхідних. Автокодувальник чи автоасоціатор — це графічна модель, яка намагається реконструювати свої вхідні сигнали.

2.2.11. RNN мережі

Невдача ANN і CNN у прогнозуванні транспортних потоків переважно пов'язані з просторово-часовим характером трафіку. Тому RNN та її форми, широко використовуються для прогнозування транспортних потоків. Однак стандартні RNN не можуть вирішити проблеми, що вимагають довгострокових тимчасових залежностей, через проблему зникаючого градієнта і градієнта, що вибухає. Для вирішення цієї проблеми використовувалася модель, яка називається рекурентною нейронною мережею з довготривалою короткочасною пам'яттю (LSTM RNN)

Основними завданнями LSTM RNN є моделювання довгострокових залежностей та визначення оптимальних часових затримок для завдань часових рядів.

Однак важко вирішити надзвичайно довгострокові залежності, можливо тому, що помилки LSTM збільшуються в міру збільшення довжини послідовності. Було запропоновано модель LSTM+, яка може відчувати як тривалу короткочасну пам'ять, так і великі відстані.

Для багатьох завдань обробки послідовності корисно аналізувати майбутнє, і навіть минуле цієї точки серії. Однак звичайні рекурентні нейронні мережі призначені для аналізу даних в одному напрямку лише в минулому. Найкращим підходом є двонаправлені мережі, розроблені Шустером і Балді.

2.2.12. Гібридні мережі

В області прогнозування транспортних потоків з'явилися гібридні моделі, спрямовані на подолання недоліків сучасних моделей, а також використання їх переваг.

Мережа прогнозування транспортних потоків (TFFNet) була запропонована для прогнозування короткострокових транспортних потоків. TFFNet складається з двох компонентів: один для просторово-часових залежностей, а інший для зовнішніх впливів.

Досить нетрадиційна гібридна модель використовує 4 секції: локальний предиктор, глобальний предиктор, інтегратор пророкування, предиктор викидів. Це показано малюнку 2.11. Локальний предиктор зумів врахувати тимчасові чинники, адаптувавши метод лінійної регресії. Глобальний предиктор враховує просторові чинники, використовуючи неконтрольовану мережу байесов для прогнозування обсягу трафіку з використанням даних з сусідніх доріг. Рівняння умовної ймовірності використовуються для заданих даних для ітеративного розрахунку значень таких параметрів, як час, тип дороги, дані та особливості точок інтересу, які визначають інтенсивність трафіку.

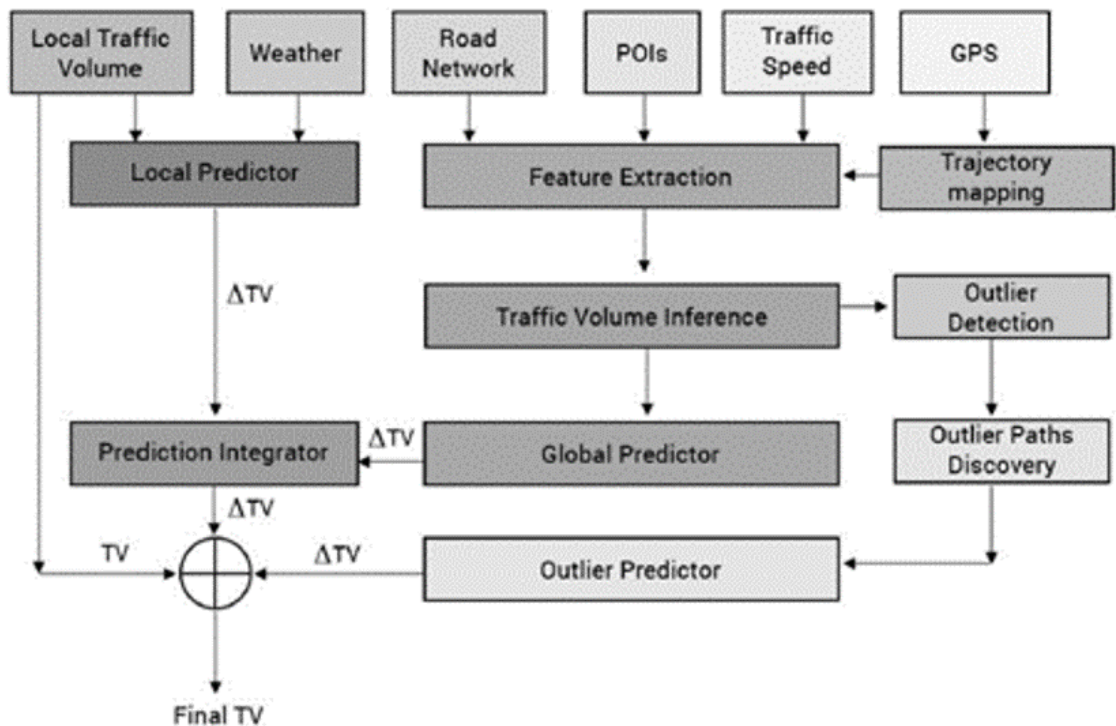


Рис. 2.11.Нетрадиційна гібридна модель[19]

Нижче у таблиці 2.1 наведено порівняльний аналіз різних мереж, їх сильні та слабкі сторони у використанні для певних цілей. Цей аналіз дає нам можливість звужити коло вибору для проекту.

Таблиця 2.1

Порівняння моделей нейромереж [19]

ТИП	СИЛЬНІ СТОРОНИ	НЕДОЛІКИ
ANN	Може виконувати нелінійну апроксимацію. Не робить припущень щодо поширення даних.	Не вдається вирішити складні проблеми, не може отримувати послідовну інформацію у вхідних даних, яка потрібна для роботи з даними послідовності. Гradient зникає або зростає, коли він поширюється назад, що призводить до зникнення та збільшення градієнта.

Таблиця 2.1 (продовження)

Порівняння моделей нейромереж [19]

ТИП	СИЛЬНІ СТОРОНИ	НЕДОЛІКИ
WNN	Вимагає меншого часу навчання порівняно з багат шаровим перцептроном. Швидка конвергенція. Відмовостійкість. Частотно-часова локалізація вейвлет-перетворення	Труднощі вибору материнської вейвлет-функції. Відсутність базису при виборі параметрів
CNN	Висока точність у проблемах розпізнавання зображень. Невелика залежність від попередньої обробки.	Для отримання оптимальних функцій потрібна велика кількість навчальних даних Дуже складний процес виявлення об'єктів
RNN	Може моделювати залежність від часу в даних. Кожен шаблон можна вважати залежним від попередніх на відміну від. Можна використовувати для динамічних задач	Для оновлення потрібно кілька параметрів. Страждає від вибухового градієнта

Таблиця 2.1 (продовження)

Порівняння моделей нейромереж [19]

ТИП	СИЛЬНІ СТОРОНИ	НЕДОЛІКИ
SAE	Зменшує розмірність даних, які використовуються. Чудово підходить для виділення функцій	Автоматичний кодер вчиться фіксувати якомога більше інформації, а не якомога більше відповідної інформації. Висока трудомісткість через багаторазовий проход вперед
HYBRID	Найкраще справляється з просторово-часовими кореляціями, оскільки поєднує в собі переваги як CNN, так і LSTM	Незважаючи на те, що гібридні мережі, які поєднують переваги як CNN, так і LSTM, здаються найефективнішим підходом для прогнозування потоку трафіку в деяких випадках не змогли перевершити мережі LSTM.

Моделі глибокого навчання для прогнозування трафіку показали перспективні результати, що відображають нелінійність прогнозування транспортних потоків. Хоча використання моделей глибокого навчання для індивідуального прогнозування потоку трафіку має кілька переваг, є істотні недоліки. Таким чином, останнім часом дослідники починають переходити від архітектур глибокого навчання до гібридних та неконтрольованих методів.

Висновки до розділу 2

У рамках даного розділу були коротко розглянуті основні підходи до аналізу даних, які необхідні для створення інтелектуальної транспортної системи. Проведена систематизація певних досліджень у галузі застосування нелінійних методів аналізу, побудованих на базі нейронних мереж, для використання в якості інструменту прогнозування транспортних потоків.

Для створення моделі способу управління транспортним потоком, було запропоновано:

1. Використання способів нелінійного аналізу привів до висновку, що необхідно використати різні алгоритми аналізу (LSTM, DQN, RF, SVR, та інші) враховуючи динамічність системи та різноманітність станів. Для цього пропонується вибрати методи машинного навчання для програмування та динамічного адаптування системи світлофорів на базі Q-learning алгоритму та застосування DQN мережі. Для аналізу трафіку та побудови моделей транспортних потоків запропоновано алгоритми LSTM, RF та SVR, з можливістю розгляду і інших моделей, враховуючи значну кількість гібридних моделей та теоретичних розробок.

2. Використання модуля аналізу та прогнозування транспортних потоків також з використання методів машинного навчання для вивчення основних транспортних потоків, вузьких міст шляхів, та планування маршрутів на базі динамічного перегляду маршрутів та використання цього аналізу для планування напрямів дорожнього руху та його перенаправлення.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ СПОСОБУ УПРАВЛІННЯ ТРАНСПОРТНИМ ПОТОКОМ З ВИКОРИСТАННЯМ НЕЙРОМЕРЕЖІ

3.1. Опис запропонованого способу управління транспортним потоком

Запропонований спосіб передбачає багатоступеневу систему, яка базується на дослідженнях оптимальних способів керування світлофорами, прогнозування заторів, управління трафіком на перехрестях, інтелектуальною системою обрахунку маршруту та системи розподілення потоків, та аналізом альтернативних архітектур побудови та планування руху шляхами за допомогою графів.

Для обрахунку результатів роботи алгоритмів та демонстрації, використано систему симуляції SUMO. Для створення симуляції використовувались карта міста Київ, а саме окремі перехрестя та розв'язки, моделі перехресть, та інші доступні моделі.

В цілому архітектура проекту передбачає використання реальних даних, щодо руху транспортних потоків у м. Київ, але за браком таких даних у відкритому доступі, запропоновано використання даних симуляції руху з певними налаштуваннями.

Таким чином реалізацію управління транспортним потоком включає наступні модулі:

- Управління світлофорами за допомогою алгоритмів машинного навчання;
- Збір даних дорожнього руху (метрики, з використанням сенсорів, або відео);
- Моделювання та побудова прогнозів дорожнього руху за допомогою нейромереж передбачувати трафік, з урахуванням обмежуючих факторів (погодні умови, ремонтні роботи, аварії, масові акції, та інше);
- Інформування, щодо альтернативності маршрутів руху у реальному часі з урахуванням GPS положення транспортного засобу, або

- інформативно на пристрої інформаційні. В даному модулі застосовуються традиційні алгоритми обрахунку маршрутів (Дейкстри, A*, Максимального потоку, тощо) – ця частина поза межами проекту ;
- Проектування альтернативних потоків (направлення руху) з урахуванням завантаженості та пропускної здатності певних ділянок, на базі історичного аналізу попиту за допомогою алгоритмів максимального потоку та з використанням нейронних мереж та симуляції транспортних потоків альтернативних напрямів руху (цей розділ за межами даної роботи)

Як приклад, на рисунку нижче представлена приблизна архітектура такої системи у загальному вигляді. Окремі елементи реалізовані модулями програми.

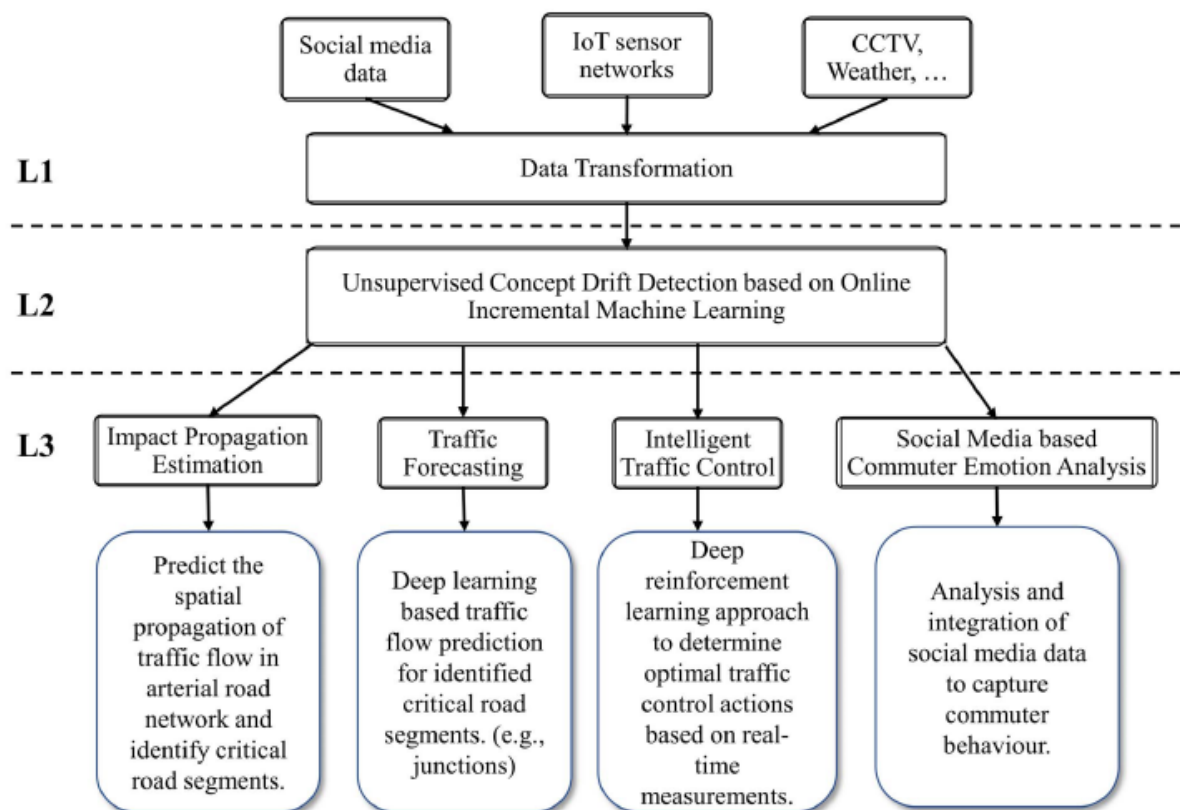


Рис.3.1 Архітектура розумної транспортної системи[21]

Після проведення огляду різних наборів послуг архітектур ARC-IT, FRAME та ISO було вибрано кілька елементів для побудови відповідної архітектури для ІТС. В архітектурі ARC-IT областю обслуговування, яка

найкраще підходить для ІТС є область управління трафіком. Після перегляду послуг, запропонованих у цій сфері, було розглянуто послуги контролю сигналів дорожнього руху і послуги вимірювання дорожнього руху, оскільки система повинна вимірювати потік транспорту на перехресті, щоб визначити найкращий час для їхніх сигналів світлофора, таким чином зменшуючи транспортний засіб час очікування. Як приклад такого схематичного представлення в форматі ARC-IT зображено на рисунку 3.2.

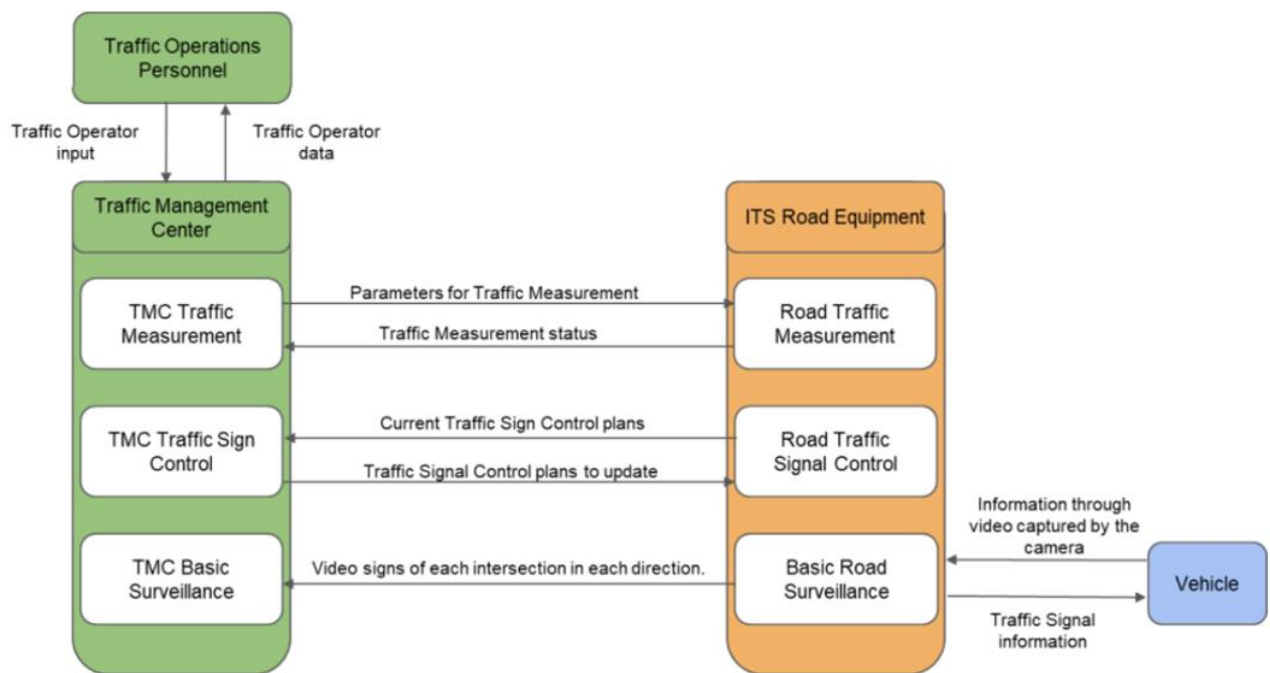


Рис.3.2 Архітектура проекту в форматі ARC-IT [23]

Функціональний модуль «Відео спостереження» (біле поле на малюнку 3.2) відповідає за захоплення та передачу відеосигналів з кожного з напрямків певного перехрестя до ТМС. Цей функціонал дозволяє візуалізувати рівень відеозавантаженості певного перехрестя за сигналами світлофора. Функціональний модуль «Управління рухом» перетворює відеосигнали в дані для визначення довжини черги транспортних засобів у кожному напрямку перехрестя. Ці дані також надсилаються до ТМС. Нарешті, функціональний модуль, визначений як «Управління світлофорами», контролює програмування відповідних фаз сигналів світлофорів на певному перехресті.

На рисунку 3.3 представлено основні модулі, що використовуються для реалізації способу управління транспортним потоком.



Рис.3.3 Схема основних модулів, які розглянуто в способі регулювання транспортного потоку

Для кожного із запропонованих модулів пропонується окремий підхід, пов'язаний з особливостями їх роботи та вимогами, щодо можливості миттєво реагувати на зміну стану (завантаженості транспортних вузлів).

Модель управління транспортним потоком передбачає використання тестових або реальних даних транспортних потоків певного міста (Київ), з аналізом даних за допомогою неймереж. Отримані результати використовуються, як прогнозна модель завантаженості певних шляхів та є входними даними для побудови альтернативних маршрутів проїзду, та окремо є входними даними для системи управління світлофорами, а саме їх часових налаштувань. Крім того, ця інформація є входною для системи попередження та

контролю швидкості по полосах руху. У Києві встановлено 7975 регульованих світлофорів, понад 700 з яких підключені до автоматизованої системи регулювання дорожнього руху. Залежно від завантаженості автомагістралей, фахівці ЦДР можуть включати на одній ділянці до восьми різних координаційних планів, які відповідають ранковим, обіднім, вечірнім годинам пік і міжпіковим періодам. Така система отримала назву Green Wave, і планувалося, що впровадження такого алгоритму впорядкує рух транспорту та збільшить пропускну спроможність доріг столиці на 20-25%. Тому основний акцент зроблено на системі управління світлофорами.

3.2. Реалізація способу управління потоками за допомогою світлофорів

3.2.1. Спосіб реалізації на базі локального агента (світлофору) – управління на базі визначення завантаження перехрестя

Перший рівень модульного способу управління транспортними потоками включає модуль який включає в себе елемент (камера або датчики) розпізнавання завантаженості шляхів (на перехрестях) та автоматичного регулювання за визначеним пріоритетом. На рисунку 3.4 представлено Т-перехрестя, як приклад моделювання



Рис.3.4 Приклад перехрестя та керування світлофорами вул. Хрещатик та Прорізна

Запропонована система отримує зображення (або дані з інших датчиків) з камер відеоспостереження на транспортних розв'язках, як вхідні дані для розрахунку інтенсивності руху в реальному часі за допомогою обробки зображень і виявлення об'єктів. Цю систему можна розділити на 3 модулі: модуль виявлення транспортних засобів, алгоритм перемикання сигналів і модуль моделювання. Зображення передається в алгоритм виявлення транспортного засобу, який використовує YOLO (натренована модель розпізнавання об'єктів) Додаток Г. Визначається кількість транспортних засобів кожного класу, таких як автомобіль, велосипед, автобус і вантажівка, для розрахунку щільності руху. Алгоритм перемикання сигналів використовує цю щільність, щоб встановити таймер зеленого сигналу для кожної смуги. Час червоного сигналу оновлюється відповідно. Час зеленого сигналу обмежено максимальним і мінімальним значенням, щоб уникнути блокування певної смуги.

Спосіб розраховано на реалізацію за допомогою систем Ардуіно, Raspberі, або інших достатніх для навчання, розпізнавання завантаженості шляхів та можливості адаптації у реальному часі тривалості певних сигналів світлофору.

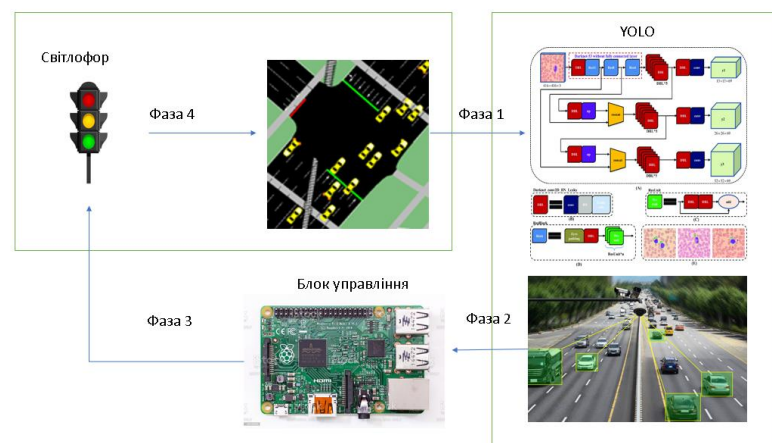


Рис.3.5 Приклад технічної реалізації системи розпізнавання та керування за допомогою мікрокомп'ютера або мікрочіпів та актуаторів (датчиків)

Фаза 1.

Отримайте зображення кожної смуги в реальному часі. Камера надсилає зображення через регулярні короткі проміжки часу до нашої системи.

Фаза 2.

Сканування та визначення щільності трафіку. система визначає кількість автомобілів на смузі руху і, отже, обчислює її відносну щільність відносно інших смуг.

Фаза 3

Введіть ці дані в модуль розподілу часу. Модуль розподілу часу отримує вхідні дані (як щільність трафіку) від цієї системи та визначає оптимізований та ефективний часовий інтервал.

Фаза 4

Результатом будуть часові інтервали для кожної смуги відповідно. мікропроцесор передає це значення на відповідний світлофор.

Розумна система світлофорів, яка використовує дані датчиків, зв'язок і автоматизовані алгоритми, була розроблена для більш плавного руху транспорту. Метою було оптимально контролювати тривалість зеленого для конкретного світлофора на перехресті. Сигнали світлофора не повинні блимати постійно одним і тим же зеленим світлом, а повинні залежати від кількості присутніх автомобілів. При інтенсивному русі в одному напрямку зелене світло має горіти довше; менше трафіку має означати коротші зелені.

Система має два налаштування – базове – фіксована тривалість (дефолт установка). Та система автоматичної адаптації до завантаженості, на основі навчання та розпізнавання об'єктів (база YOLOv3). На рисунку показаний аналіз ефективності розпізнавання об'єктів [24].

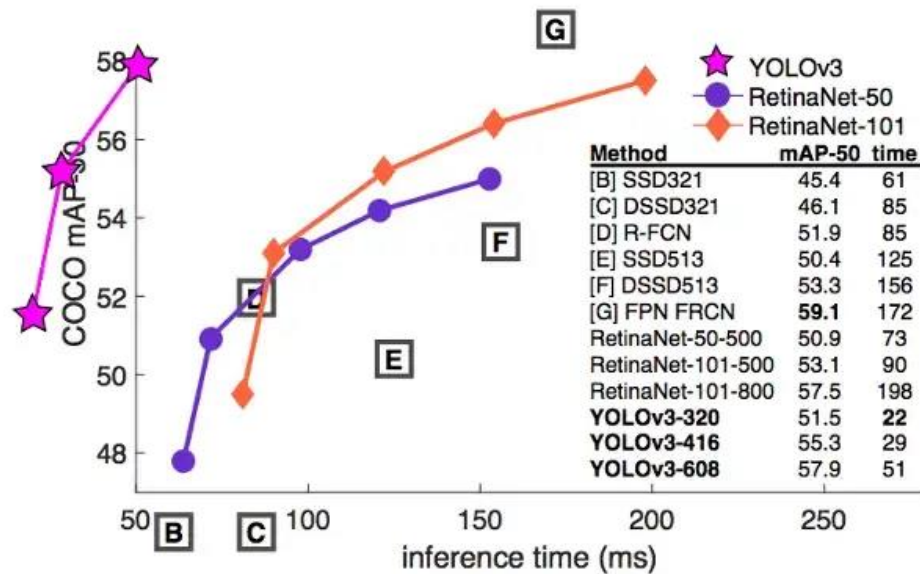


Рис.3.6 Порівняльний графік ефективності YOLO [24]

YOLOv3 досягає найкращого співвідношення швидкості та точності на наборі даних MS COCO, великомасштабному наборі даних виявлення об'єктів..

YOLO версії 3 — це система виявлення об'єктів у реальному часі, яка, на відміну від інших систем, таких як згорткові нейронні мережі на основі регіону (R-CNN), застосовує одну нейронну мережу до всього зображення. Ця мережа ділить зображення на регіони та генерує обмежувальні рамки та ймовірності для кожного регіону, що робить цю систему в 100 разів швидшою за Fast R-CNN.

Модель тренували за допомогою попередньо підготовлених ваг, завантажених із веб-сайту YOLO. Конфігурацію файлу .cfg, який використовується для навчання, було змінено відповідно до специфікацій моделі. Кількість вихідних нейронів на останньому шарі було встановлено рівним числу класів, які модель повинна виявити шляхом зміни змінної 'classes'. У нашій системі це було 4, а саме. Автомобіль, велосипед, автобус/вантажівка та мотоцикл. Кількість фільтрів також потрібно змінити за формулою $5 \cdot (5 + \text{кількість класів})$.

Фактично програма керує фазами роботи світлофора, які представлено на рисунку знизу.



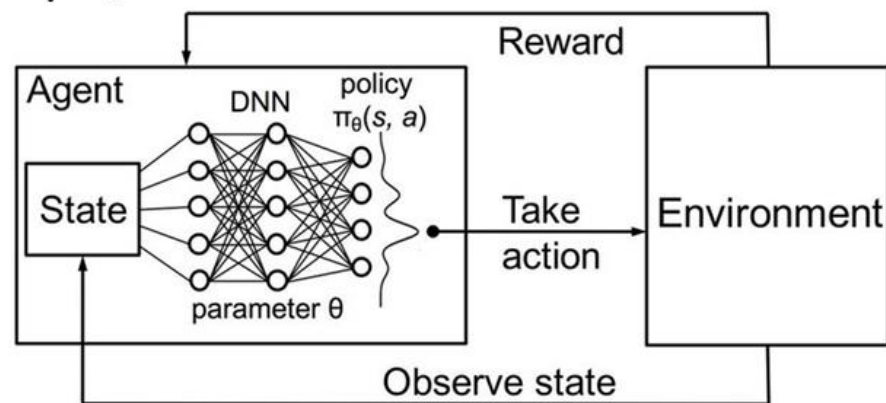
Рис.3.7 Фази роботи світлофора

Цей спосіб є простим в плані реалізації та ефективним з точки зору надійного простого рішення локального рівня. Але він не бере до уваги ситуацію на інших перехрестях.

3.2.2. Спосіб реалізації на базі RL (reinforcement learning) (світлофору)

Метою алгоритму RL є мінімізація довжини черги транспортних засобів на перехрестях, для чого встановленою винагородою є зменшення черги транспортних засобів у кожному з напрямків. Дія, виконана агентом, генерує новий стан у сигналах світлофора (новий час фази). На рисунку 3.8 представлено узагальнене коло навчання за Q алгоритмом.

Deep Q-Network



Ref: URL: <https://2.bp.blogspot.com/-bZERYUNyjao/Wa98yt7Gjhl/AAAAAAACT8/SYQjUNrbe1YDtKTMKR6LPt68C0pPqkoowCLcBGAs/s1600/DRL.JPG>

bZERYUNyjao/Wa98yt7Gjhl/AAAAAAACT8/SYQjUNrbe1YDtKTMKR6LPt68C0pPqkoowCLcBGAs/s1600/DRL.JPG

Рис.3.8 Коло навчання мережі в моделі RL

Цей новий стан, можливо, збільшить або зменшить розмір кожної черги транспортних засобів. Алгоритм має на меті зменшити довжину всіх довгих черг із новим призначеним станом, отже, одержуючи найвищу винагороду. Якщо цього не станеться, отримана винагорода буде змінюватися, відповідно до встановленої політики.

3.2.2.1. DQN використання в програмі

Q-навчання – це алгоритм навчання з підкріпленням. Алгоритм оцінює остаточну винагороду в кожному стані замість досягнення кінцевого стану, відстеження та оновлення функції значення. Це дає можливість використовувати Q-навчання в середовищі без кінцевого стану

Це навчання буде виконано лише під час першого виконання алгоритму; під час кожного запуску алгоритму Q-значення оновлюються, тому алгоритм постійно навчається.

Після завершення навчання починається робота з регулювання дорожнього руху. Агент піддається впливу навколишнього середовища та отримує різні винагороди, виконуючи різні дії. Значення Q оновлюються за допомогою наступного рівняння :

$$Q^*(st, ac) \leftarrow (1-\alpha)Q(st, ac) + \alpha(rw + \max Q(\text{наступний стан, усі дії}))$$

де $Q^*(st, ac)$ — нове Q -значення, $Q(st, ac)$ — поточне Q -значення для пари стан–дія, α ($0 < \alpha \leq 1$) — швидкість навчання, rw — винагорода, отримана від навколишнього середовища, а γ ($0 \leq \gamma \leq 1$) — ставка дисконту, яка визначає, наскільки важливі майбутні винагороди. Параметри α (альфа) і γ (гама) відомі як «гіперпараметри» в алгоритмах Q -навчання.

На рисунку 3.10 представлена блок-схема, що пояснює роботу алгоритму.

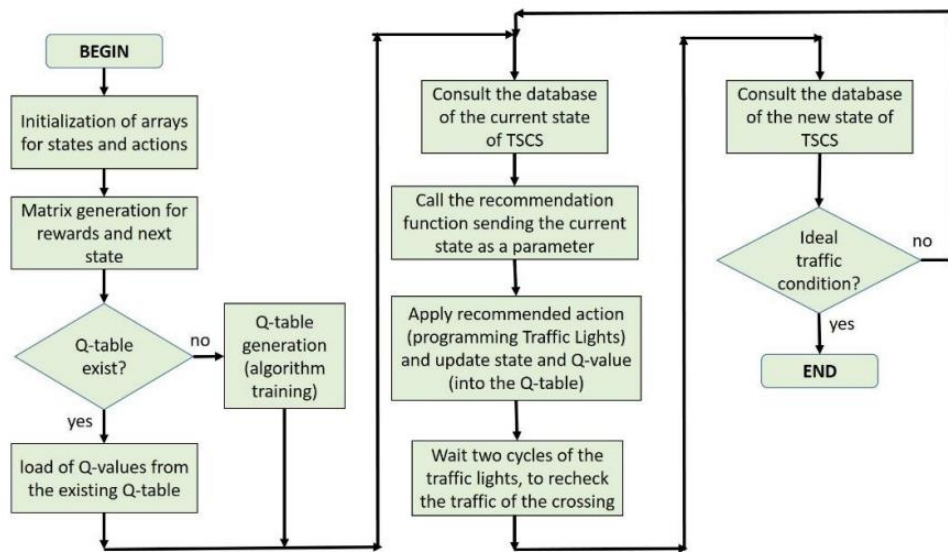


Рис.3.10. Блок-схема алгоритму RL [23]

Оскільки неможливо було протестувати систему інтегровано з усіма модулями в реальному середовищі, було запропоновано розробку моделювання за допомогою відповідного програмного забезпечення (SUMO). Приклад алгоритму DQN представлено з реалізацією на SUMO на рис.3.11

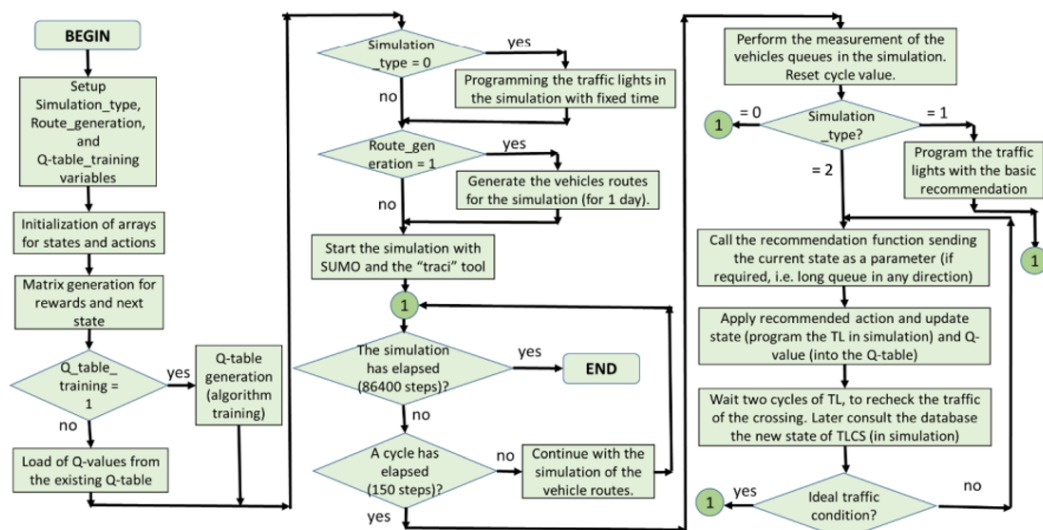


Рис.3.11 Алгоритм DQN з використанням симуляції в SUMO[23]

Архітектура псевдокоду алгоритму DQN представлено в Додатку Д.

Основні підходи до керування світлофорами поділяють на централізоване, кластерне та індивідуальне. Основна проблема централізованої системи – це проблема розмірності, яка зростає з додаванням агентів у систему навчання, що створює обчислення доволі складним та довготривалим, що не є прийнятним якщо потрібна робота у реальному часі. На рисунку 3.12 зображена проблема централізованого управління, пов'язана з поняттям Прокляття розмірності та проблемами обчислення станів.

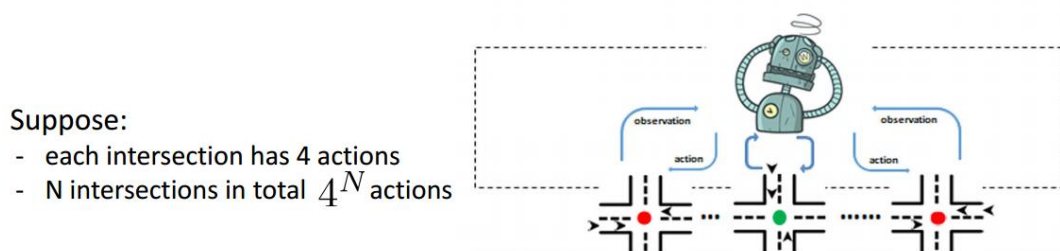


Рис.3.12 Проблема розмірності у централізованій системі управління світлофорами[25]

Кластерна (координація з найближчими сусідами) система є більш практичною для реалізації та розглядається як можливий варіант розвитку програмного продукту, але в даному випадку не використовувалась. На рисунку 3.13 представлена концепція координації дій сусідніх агентів (світлофорів). Як показують дослідження, ця система є більш ефективною ніж індивідуальна.

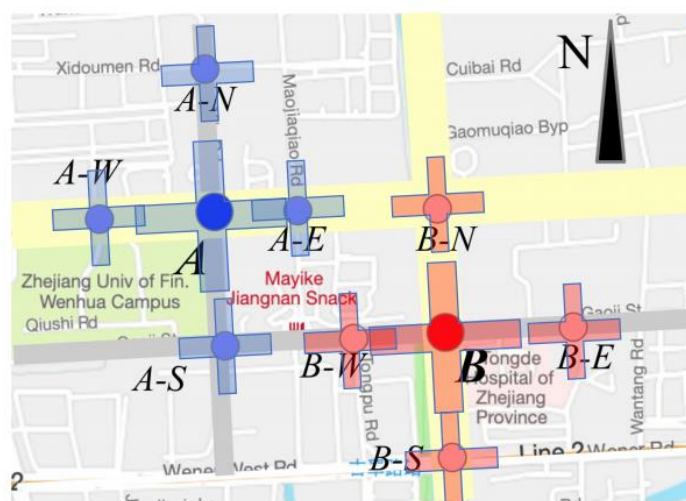


Рис.3.13 Кластерна система керування світлофорами[25]

Для реалізації програми керування світлофорами з навчанням за алгоритмом DQN було обрано індивідуальну модель, оскільки з технічної точки зору це простіше для реалізації та демонстрації. На рисунку 3.14 представлена схематично така система.

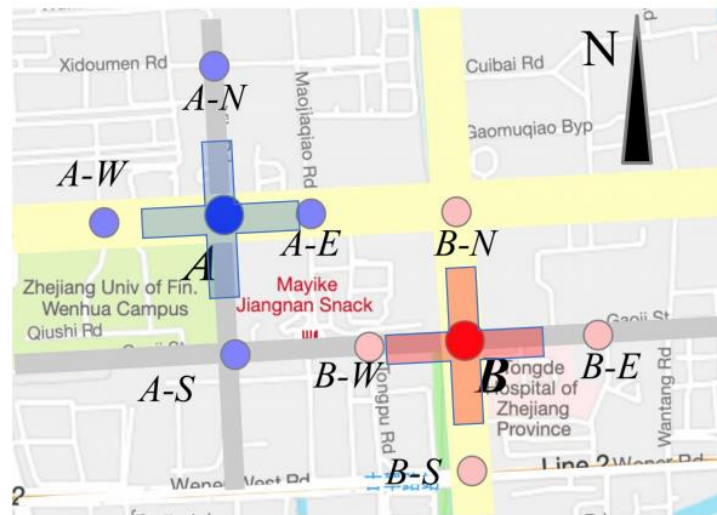


Рис.3.14 Індивідуальна система керування світлофором[25]

Архітектура модуля тренування та прогнозування руху на перехресті та тренування роботи світлофора, відповідно до моделі завантаженості.

Система управління світлофором, реалізована за допомогою навчання з підкріпленням. Для подальшого покращення ефективності навчання DQN інтегровано градієнтний алгоритм метанавчання. Він належить AutoRL і може регулювати параметр гамма під час навчання. Експерименти показують, що це сприяє конвергенції функції Q-value і певною мірою уникає переоцінки, особливо у випадку невідповідних налаштувань навчання, тоді як DQN зазнає невдачі в навчанні.

Експерименти призначені для порівняння продуктивності навчання з використанням різних алгоритмів (DQN) з різними оптимізаторами (SGD, RMSprop).

Модель завантаженості формується шляхом симуляції станів (State) з використанням модуля SUMO-GUI. Налаштування моделюються на певну геометрію. В принципі, можна використати реальні мапи, та архітектуру побудовану відповідно на геометрії та реальних потоках транспорту. Для такої побудови необхідно визначити статистику руху за певними маршрутами, з

відповідним навантаженням по годинам. Але для моделювання руху та аналізу можна використати гіпотетичну модель мікроруху на окремому перехресті.

Головним модуль програми `main.py` відповідає за завантаження агента навчання, в даному випадку це Q-learning алгоритм та запуску середі симуляції `sumo-gui`. Фактично модуль навчання аналізує рух на перехресті та визначає оптимальний варіант регулювання (послідовності перемикання сигналів світлофору). Всі основні налаштування моделі навчання, можуть змінюватись за допомогою флагів (які фактично виконують роль API).

Основні флаги моделі:

Кількість епізодів - ('num_episodes', 601, ")

Агент навчання - ('network', 'dqn', ")

Визначення моделі тренування чи навчання ('mode', 'train', ")

Визначення оптимізаторів('use_sgd', True, 'Training with the optimizer SGD or RMSprop').

Такими чином програма виконує наступні фази:

Фаза 1 – Запуск налаштованого середовища SUMO та генерація руху транспорту. Роботу світлофорів. Дані щодо статистики руху всіх авто та витраченого часу на проходження перехрестя за умов затримки на світлофорах.

Фаза 2 - На основі цих даних – агент навчання – світлофор – тестує різні фази роботи в залежності від кількості ваг та винагороди – визначає оптимальний варіант роботи. Функція винагороди в даному випадку це мінімізація витрат часу всіма транспортними засобами, що проходять перехрестя. На рисунку нижче представлений фрагмент навчання моделі.

```
Step #1001.00 (0ms ?*RT. ?UPS, TraCI: 4ms, vehicles TOT 743 ACT 54 BUF 13)
i_episode: 598
eps_threshold = : 0.8346946746653232
learn_steps: 121280
gamma: tensor(0.9001)
Step #1001.00 (0ms ?*RT. ?UPS, TraCI: 7ms, vehicles TOT 746 ACT 81 BUF 10)
i_episode: 599
eps_threshold = : 0.834446867660534
learn_steps: 121776
gamma: tensor(0.9001)
```

Рис.3.15 Навчання моделі

Фаза 3 – формування ваг тренування та можливість використання налаштованої моделі для регулювання перехрестя у децентралізованій системі.

В результаті навчання ми отримуємо результати навчання моделі та ваги отримані після навчання можна використовувати для роботи світлофора в режимі більш ефективному ніж, наприклад фіксований світлофор.

Показник ефективності в даному випадку – час витрачання на чергу.

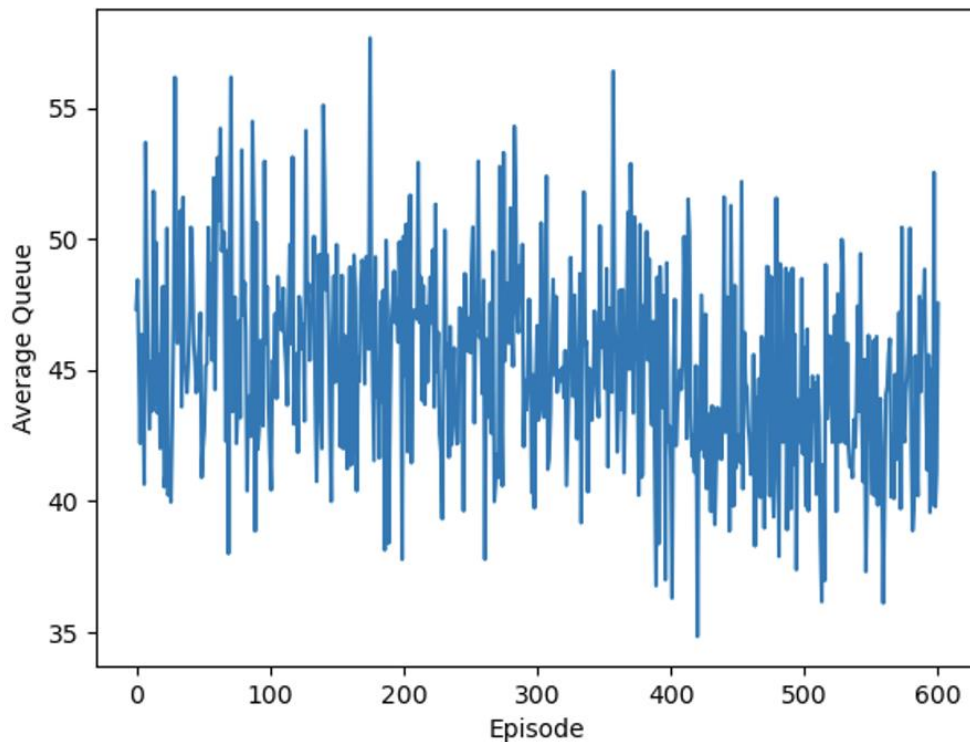


Рис.3.16 Результат навчання мережі

Після навчання мережі на конкретних прикладах, можна застосувати ваги для реалізації у конкретних умовах. В даному випадку для демонстрації результату використовуємо той же симулятор SUMO.

Таким чином у цьому модулі ми реалізуємо модель навчання системи керування світлофором використовуючи не показники аналізу завантаженості певних напрямків та наданням пріоритету завантаженим, шляхом аналізу відео за допомогою алгоритмів розпізнавання YOLO чи інших, або за допомогою використання датчиків, а за допомогою методів нелінійного аналізу та пошуку, оптимального часу перемикання фаз світлофора. Хоча застосування навчання за допомогою того ж Q- learning (DQN) алгоритму і вважається ефективним, все ж його застосування на окремому перехресті є обмеженим. На рисунку 3.17

показано моделювання вже на базі навченого агента (світлофора) шляхом передачі даних симулятору SUMO.

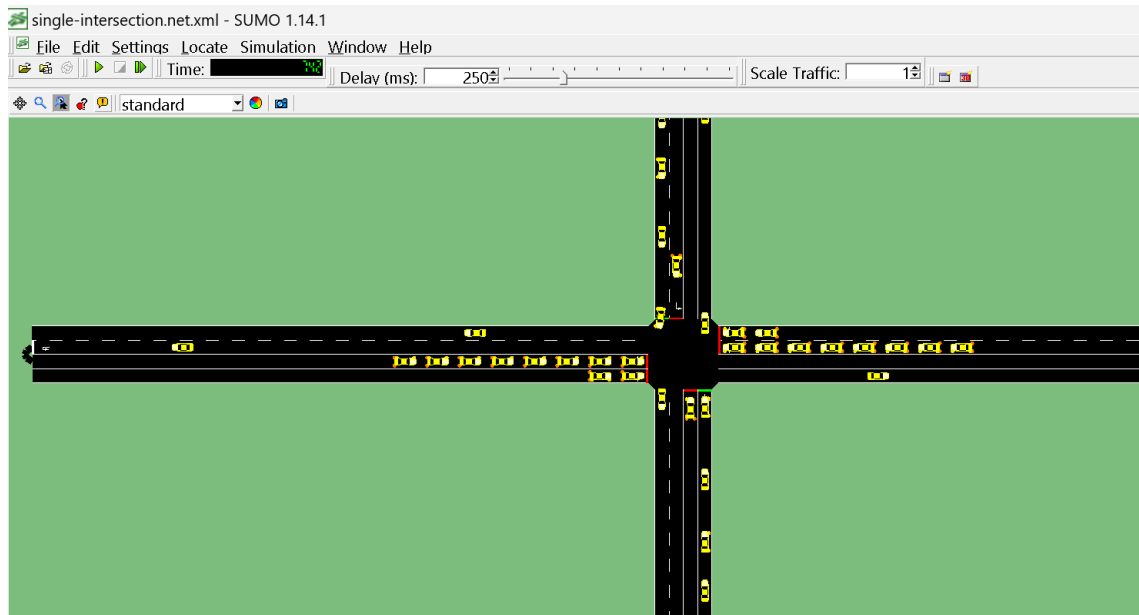


Рис.3.17 Демонстрація навченого агента за DQN алгоритмом

Але основне значення має порівняння застосованого алгоритму навчання з базовим варіантом – без застосування навчання. Статистика яку надає симулятор SUMO надає можливість порівняти «Час втрат на черги».

Наприклад на рисунку 3.18 нижче показано порівняння застосованої моделі- умовно від базового варіанту до найкращого за умов певних налаштувань та тренувань мережі.

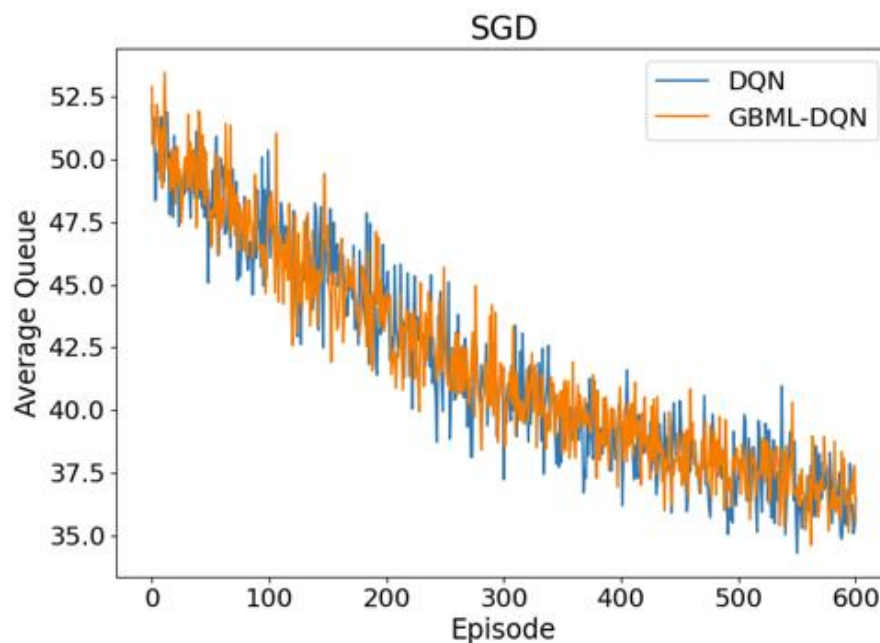


Рис.3.18 Результат навчання та вплив на Час витрачання у черзі

Таким чином ми має другий шар способу інтегрованого управління трафіком. Наступний рівень стосується аналізу трафіку у макро та мезо масштабах, а саме аналіз руху на рівні самого міста, певних ділянок доріг.

3.3. Спосіб управління потоками за рахунок прогнозування транспортних потоків та їх перенаправлення

Для реалізації блоку аналізу транспортних потоків та симуляції також використовується модель: Програмний блок, на базі скрипту на мові Python та транспортна статистика, яка надає дані для аналізу. В даному випадку моделювання даних статистики дає можливість проводити аналіз руху як на певних ділянках, перехрестях, або певних маршрутах (які можна розглядати, як графи), та використати такі ділянки для аналізу. В даному випадку для навчання моделей було обрано декілька алгоритмів, оскільки є можливість порівняти застосування різних підходів. В даному випадку використано 5 основних алгоритмів: LSTM, GRU, SAEs, Random Forrest та SVR. На рисунку 3.19 схематично зображено будову алгоритмів. Структура представлена в Додатку Д.3

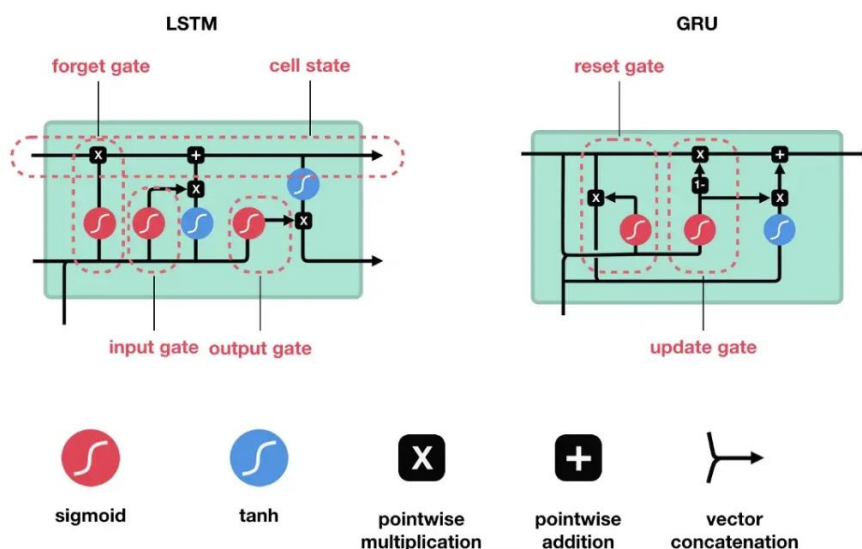


Рис. 3.19 Схема алгоритмів LSTM та GRU

Даний модуль складається також з модуля навчання `train.py` та модулю виконання – `main.py`. Модель отримує данні з файлів у форматі `scv`. Які поділено

на групу навчання та тестування. На рисунку 3.19 представлений результат навчання та метрики відповідно до застосованих алгоритмів навчання.

```
LSTM
explained_variance_score:0.941023
MAPE:16.895268%
MAE:7.302074
MSE:97.678800
RMSE:9.883259
R2:0.939869
GRU
explained_variance_score:0.941564
MAPE:17.668590%
MAE:7.196727
MSE:97.318209
RMSE:9.864999
R2:0.940091
SAEs
explained_variance_score:0.944269
MAPE:17.801078%
MAE:7.058258
MSE:92.077206
RMSE:9.595687
R2:0.943318
```

Рис.3.20 Метрики навчання та показники помилок за моделями

В результаті роботи ми отримали навчання мережі – ваги. Які використали для тестування роботи мереж. Результат тестування представлений на рисунку 3.21.

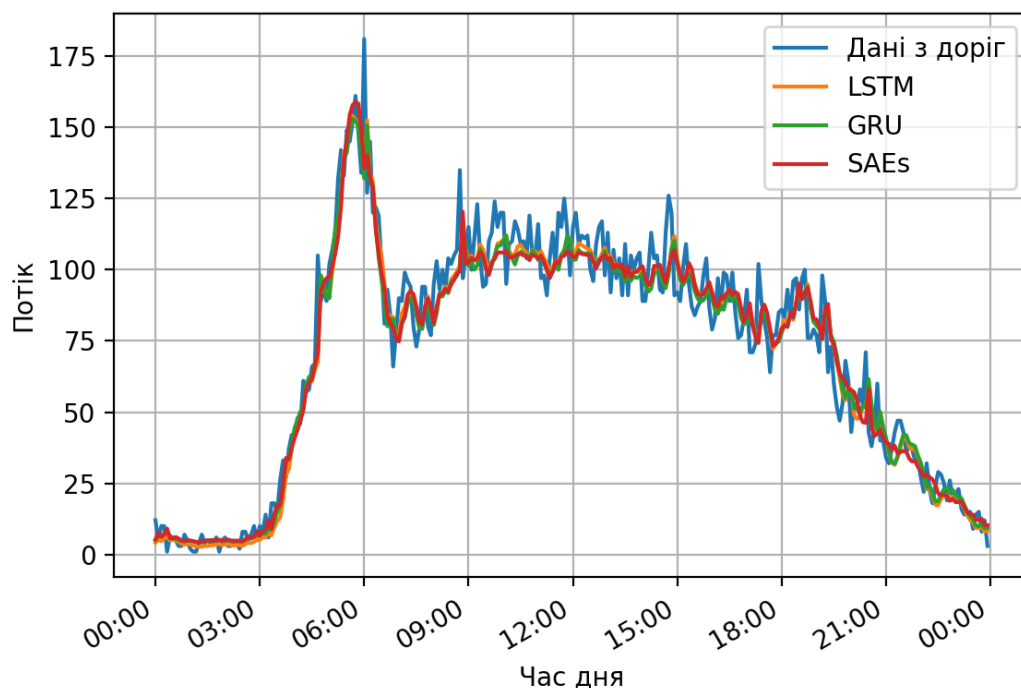


Рис.3.21 Ефективність відтворення прогнозу транспортного потоку

Таким чином третій модуль системи працює для створення прогнозів даних щодо завантаження певних ділянок, або міста, або маршрутів – в залежності від статистику яку ми використовуємо. Але робота модуля має на

меті скоріш інформаційну складову ніж управління в реальному часі, оскільки є висока ймовірність помилки у передбаченні.

3.4. Моделювання та симуляція алгоритму прогнозування та моделювання транспортного потоку

В роботі для генерації даних, моделювання запропонованого способу управління та для візуальної демонстрації використовується модуль симуляції SUMO, який можна використовувати, як сервер для обробки даних щодо трафіку та динамічної зміни налаштувань. На рисунку 3.22 представлено вулицю Хрещатик, як приклад демонстрації, як приклад кластерної системи управління світлофорами.

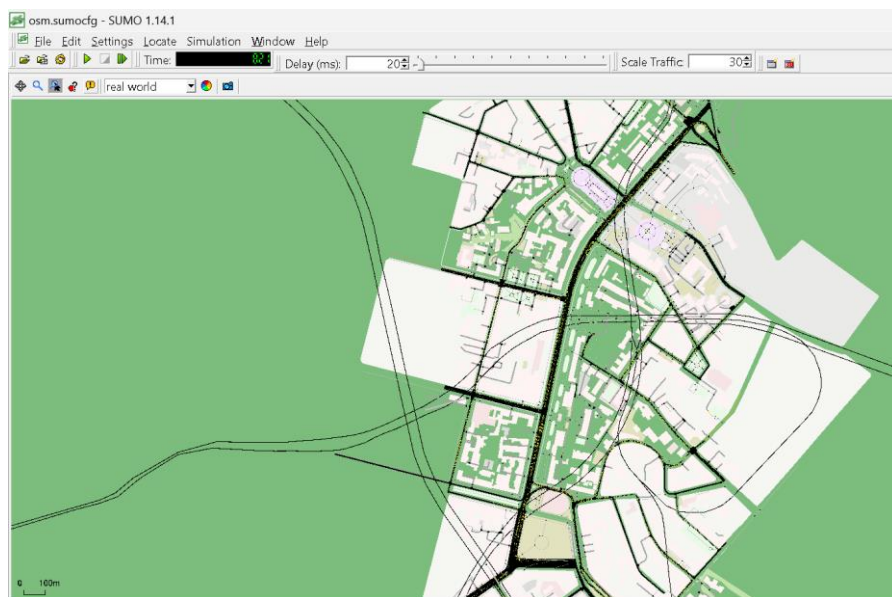


Рис. 3.22 Приклад вулиці Києва у системі SUMO

Більше детальне представлення для децентралізованого способу управління представлено на рисунку 3.23 знизу.

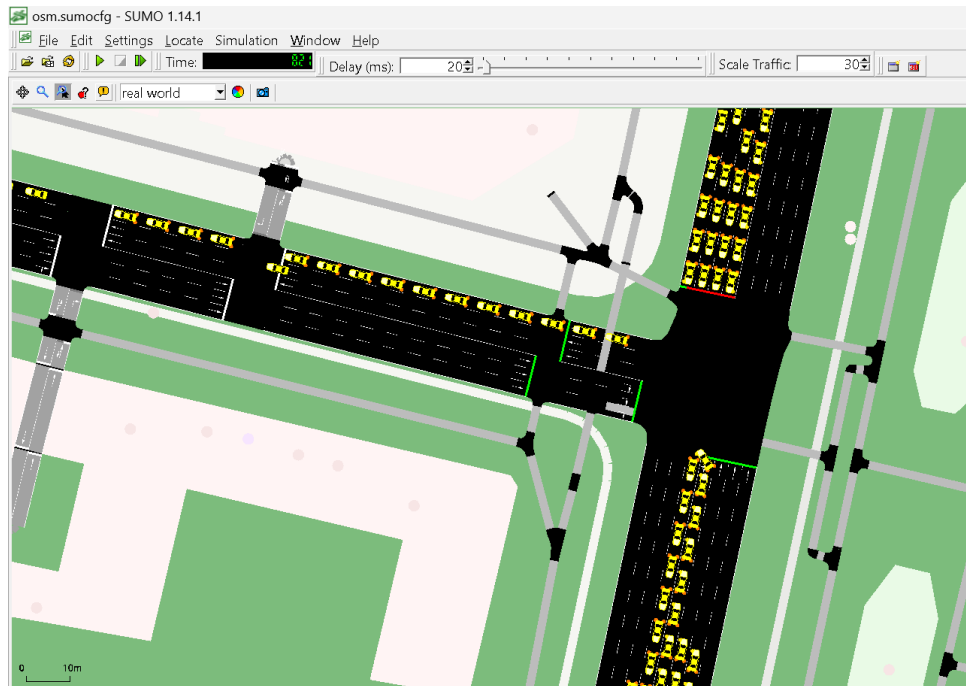


Рис. 3.23 Приклад перехрестя вул. Хрещатик з можливістю керування світлофорами з урахуванням результатів машинного навчання та аналізом статистики.

SUMO (скорочено SUMO) — це потужний симулятор, розроблений для роботи з великою мережею навантажень і заданим попитом на трафік, включаючи маршрут транспортного засобу та модель, що слідує за ним. Він також надає багато корисної інформації, такої як швидкість автомобіля, модель і місцезнаходження. Однією з головних особливостей SUMO є інтерфейс керування дорожнім рухом (або скорочено TraCI), який є API Python, який розглядає симуляцію SUMO як сервер і дозволяє користувачам отримувати інформацію з симуляції трафіку або змінювати симуляцію. TraCI забезпечує інтерфейс, що дозволяє стороннім системам (або бібліотекам) інтегруватися з симуляцією трафіку SUMO.

Навчання з підкріпленням як метод машинного навчання привело до дуже багатообіцяючих результатів як рішення для складних систем. Метод навчання з підкріпленням здатний отримати знання або покращити продуктивність шляхом взаємодії з самим середовищем. Мета навчання з підкріпленням полягає в тому, щоб вивчити оптимальну політику на основі винагороди, отриманої шляхом

повторення взаємодії з середовищем. Це послідовно оптимізує політику та, зрештою, створює рішення.

Хорошим прикладом застосування навчання з підкріпленням у реальній ситуації є управління дорожнім рухом у містах. Оскільки кількість транспортних засобів у нашій міській транспортній мережі зростає, розробка інтелектуальної системи керування дорожнім рухом для виконання інтелектуального маршрутизації для транспортних засобів є великим попитом. Однак складність міської транспортної мережі викликає багато проблем в управлінні дорожнім рухом, наприклад, швидкісні зміни систем руху та широкий розподіл транспортних засобів на проїжджій частині. Симулятор SUMO може дозволити стороннім системам підійти до навчання з підкріпленням. У цьому випадку TraCI відіграватиме роль «перетворювача» між SUMO та підходом навчання з підкріпленням для встановлення цієї взаємодії. TraCI може отримати кожен інформацію в симуляції, включаючи транспортний засіб і мережу. Це надає корисні функції агенту навчання з підкріпленням для обґрунтування станів середовища. Ґрунтуючись на спостереженні за станами, ми могли встановити та призначити винагороди відповідним чином і дозволити навчання з підкріпленням оптимізувати політику на основі винагороди. Після цього агент навчання з підкріпленням призначить нову дію для SUMO через TraCI і постійно спостерігатиме за станом середовища.

Доступ до TraCI можна отримати за допомогою кількох мов програмування, найпоширенішою з яких є Python. Пакет інструментів/TraCI у симуляторі SUMO дозволяє користувачам взаємодіяти з SUMO за допомогою Python.

Взаємодія між агентом навчання з підкріпленням і середовищем через TraCI триватиме, поки він не досягне кінцевого стану або агент не виконає умову завершення. По суті, методи навчання з підкріпленням застосовують процеси прийняття рішень Маркова (MDP). MDP визначається як п'ять кортежів $\langle S, A, T, R, \gamma \rangle$, де S – набір станів, A – набір дій, які можуть змінити статус, T – функція переходу, яка є ймовірність зміни стану під час певної дії, R є функцією

винагороди, а γ відомий як коефіцієнт дисконтування, який моделює важливість майбутніх і негайних винагород.

Навчання з підкріпленням оптимізує свою політику, повторюючи наступний крок: на кожному кроці часу t агент навчання з підкріпленням сприймає стан із колекції станів S і на основі своїх спостережень вибирає дію та виконує її, щоб привести стан середовища до переходу до наступний стан. Потім агент отримує негайну винагороду R , спостерігає за новим станом і оновлює політику за допомогою рівняння вище, включаючи знижену винагороду γ .

Щоб підійти до навчання з підкріпленням у симуляторі SUMO, нам потрібні відповідні елементи для Марківського процесу прийняття рішень у SUMO. Симулятор SUMO використовує метод маршрутизації за замовчуванням «DuaRouter» для створення файлів маршруту для кожного транспортного засобу в симуляції. «DuaRouter» виконує динамічне призначення користувача (DUA) на основі обчислення найкоротшого шляху. Агент навчання підкріплення замінить метод маршрутизації за замовчуванням оптимальною політикою, яку він навчився.

TraCI використовується для виконання навчання з підкріпленням за допомогою симулятора SUMO. На це є декілька причин. По-перше, TraCI надає користувачеві доступ до симуляції поточного руху. Таким чином, легко отримати дані змодельованих об'єктів (таких як транспортний засіб, дорога, маршрут) і маніпулювати їх поведінкою онлайн. По-друге, TraCI дозволяє керувати своїм інтерфейсом кількома мовами програмування, такими як Python, Matlab і C++, що дозволяє користувачеві легко імпортувати бібліотеку Reinforcement Learning і виконувати функції Reinforcement Learning за допомогою SUMO. Нарешті, результати моделювання SUMO надають інший набір даних для навчання та тестування за допомогою Reinforcement Learning, наприклад перевірку типу транспортного засобу (положення транспортних засобів у часі для певного типу транспортного засобу), необроблений дамп положення транспортного засобу (усі положення та швидкість транспортного засобу з часом), вихід викидів (значення викидів усіх транспортних засобів для

кожного кроку моделювання). За допомогою цих даних користувачі можуть навчити машину працювати швидше й точніше, визначаючи, як покращити керування трафіком у транспортній мережі.

Використовуючи TraCI, ми можемо імітувати затори або змінити маршрут. Затор означає все, що зупиняє рух транспорту. Прикладами в реальному світі можуть бути зіткнення, дорожні роботи, погані погодні умови, що призводять до дуже низької швидкості, або просто велика кількість транспорту. У SUMO є кілька варіантів імітації таких подій. Зупинка автомобіля на фіксований період часу, шляхом визначення точки на його маршруті, коли він повинен зупинитися та на який час. Маніпулювання світлофорами таким чином, щоб світлофор залишався червоним довше, ніж звичайна тривалість фази, або встановлення обмеження швидкості на смузі так низько, що трафік практично не змінюється.

Проте зміна маршруту означає зміну шляху, яким транспортний засіб проходить за певних умов. Традиційно маршрутизація транспортних засобів використовує статично визначений механізм маршрутизації, наданий SUMO. Цей статичний метод розгортається шляхом додавання файлу перемаршрутизації до конфігурації SUMO.

3.4.1. Результати використання способу управління транспортним потоком

В результаті роботи програми ми отримуємо два елементи управління трафіком – управління трафіком у реальному часі, а саме за допомогою алгоритму розпізнавання (YOLO) або іншої системи фіксації кількості транспорту на перехресті, ми в реальному часі визначаємо пріоритетність, чим принаймні, оптимізуємо локальну завантаженість перехрестя.

Другим кроком програми, ми навчаємо систему більш обґрунтованій моделі з використанням алгоритмів нелінійного навчання DQN, яка показує значні покращення статистики витрачання часу на перехресті (хоча і в локальному виконанні). Конкретні показники статистики треба визначати на реальних прикладах.

Ці два способи дають змогу керувати в реальному часі.

Третій спосіб доданий до системи – це передбачення завантаженості шляхом аналізу статистики руху на певних ділянках, чи в цілому по місту за допомогою методів нелінійного аналізу та застосування алгоритмів LSTM, GRU, SAE, RF, SVR. Як показав аналіз на певному наборі даних – передбачення дало 94% точності, але тестування треба проводити на реальних даних локально. Тоді можна говорити про конкретну статистику певних ділянок або територій.

Висновки до розділу 3

У рамках даного розділу були детально розглянуті аспекти архітектури програмного продукту, обґрунтовано вибір мови програмування та технологію для написання програми. Крім цього, в даному розділі було вирішено наступні завдання:

1. Була розглянута модульна архітектура (мікро сервісна) реалізації програми.
2. Було обґрунтовано вибір мови програмування Python та використання бібліотек Tensorflow, та бібліотек для розробки.
3. Були проведено аналіз вибору алгоритмів навчання нейромереж, як основного інструменту аналізу та керування.
4. Було проведене моделювання та використання симулятора SUMO для представлення результату навчання моделей.
5. Був проведений певний аналіз статистики, що до покращення руху транспорту в умовах індивідуальної реалізації на окремому перехресті.

РОЗДІЛ 4

РОЗРОБКА СТАРТ АП ПРОЕКТУ

4.1. Опис проекту

Ідея проекту полягає у створенні багаторівневої системи аналізу та управління частиною ITS (інтелектуальної транспортної системи, а саме сигнально інформаційних елементів: світлофорів та прогнозування транспортних та маршрутизація транспортних потоків. Крім того, є елементи системи управління, які відносяться до поняття проектування шляхів та реформування напрямків руху, які мають базуватись на аналізі статистики завантаженості шляхів. Для цих цілей можливе використання теорії графів та алгоритмів потоків. Таким чином ідея полягала у системному аналізі та керуванні найбільш впливовою частиною системи ITS у частині керування та активного управління трафіком в частині:

- Зменшення завантаженості шляхів;
- Зменшення заторів;
- Активного реагування на події (інциденти, соціальні події, тощо);
- Активного планування нових шляхопроводів, та перегляду моделей руху існуючими шляхами.

В рамках даної роботи, розглядались лише частина аспектів системи, а саме використання елементів штучного інтелекту (алгоритми глибинного навчання) для навчання активного реагування світлофорів та аналіз і прогнозування завантаженості шляхів або певних маршрутів, з метою активної ре маршрутизації транспортних потоків за умов їх рівномірного розподілу.

В таблиці 4.1 представлені основні аспекти ідеї можливі напрямки застосування, основні вигоди та інша корисна інформація.

Таблиця 4.1

Опис ідей стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Створення Web додатку або програми (враховуючи особливість застосування можливо Web додаток не є добрим рішенням, але клієнт серверна архітектура, як основна модель) для отримання даних актуаторів та активного провадження методів штучного інтелекту для аналізу та активного управління кластерами або індивідуальними агентами (світлофорами)	Збір статистики транспортних потоків з рахунок відео та даних датчиків	Надання інформації щодо події на шляхах, збір основних маршрутів та типових шляхів пересування для подальшого аналізу.
	Моделювання оптимальних дій системи на стани та навчання агентів (світлофорів)	Активне управління транспортними потоками для зменшення заторів та зменшення часу подорожей, економія пального та зменшення викидів CO ₂ .
	Прогнозування заторів та моделювання транспортних мереж та їх розвиток	Прогнозування місць з підвищеним навантаженням на вузли та канали задля попередження потенційних поломок та збоїв у роботі.

Таблиця 4.1 (продовження)

Опис ідей стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
	V2I розвиток мережі на базу VANET	Збільшення активного управління за рахунок автоматизованих та інтелектуальних авто (контроль швидкості, налаштування маршрутів руху, запобігання порушенню правил та аварій)
	Створення VTMS (virtual traffic management system)	Гнучка хмарна система управління транспортом.

4.2. Порівняльний аналіз конкурентних систем

В принципі важко говорити про конкретних конкурентів, оскільки даний напрямок має багато компаній як великих промислових, так і відкритих проектів. Тому поділимо їх на кластери – 1 кластер – це великі програмні продукти, 2- це середні комерційні продукти, 3 – безкоштовні та відкриті системи. Тому порівняння буде носити гіпотетичний характер з визначенням особливості даного проекту.

Таблиця 4.2

Визначення сильних, слабких та нейтральних характеристик ідей проекту

№	Техніко-економічні характеристики ідеї	Мій проект	Конкуренти		
			1	2	3
1	Модульність системи (мікро сервіси)	10	12	10	8
2	Кроссплатформенність	8	12	9	7
3	Робота як з даними в реально часі та використання їх для створення прогнозів та навчання	6	10	8	6
4	Гнучкість застосування	10	6	6	10

Визначений перелік слабких, сильних та нейтральних характеристик та властивостей ідей потенційного товару є підґрунтям для формування його конкурентоспроможності. Як видно з таблиці 4.2, основною перевагою розроблюваного продукту є саме графічне відображення компонент, на що і варто зробити ставку. В майбутньому варто врахувати основний недолік розроблюваного продукту, а саме додати віртуалізацію вузлів з можливістю вводити базові команди в консолі вузлів.

4.3. Оцінка ідеї проекту

Таблиця 4.3

Технологічна спроможність реалізації проекту

№	Ідея проекту	Технології її реалізації	Наявність технології	Доступність технологій
1	Створення модульної програми для теоретичного аналізу та симуляції з наступним впровадження моделей в реальні комерційні проекти	Мова програмування Python (але для комерційних продуктів краще використання C++, враховуючи необхідність програмування мікроконтролерів	Є в наявності (потребує доробки система роботи з мікроконтролерами, актуаторами, камерами	Доступні безкоштовно
2		Інтеграція з бібліотеками штучного навчання (Tensorflow та інші)	Є в наявності	Доступні безкоштовно
3		Алгоритми DQN, LTSM, SAES, GRU, etc	Є в наявності	Доступні безкоштовно

Обрана технологія для створення прототипу з використання мови Python та бібліотек штучного навчання є найбільш простим рішенням, враховуючи, певну складність та багатошаровість рішення. Проста і швидка інтеграція

модулів або складання модулів за принципом один модуль одне завдання дає можливість гнучко реагувати на потреби. Та не перенавантажувати моделювання – від простого до більш складного.

4.4. Аналіз оцінки можливості провадження проекту на ринку

Після аналізу продукту переходимо до аналізу ринкових можливостей проекту. Перш за все це питання визначення можливостей та загроз по типу SWOT (сильні, слабкі сторони, можливості та небезпека) – аналізу.

Подібний аналіз дає змогу на макро рівні оцінити основних конкурентів, основні слабкості та можливості та правильно позиціонувати свій проект та правильно орати стратегію розвитку, наприклад стратегію фокусування, або мінімальних витрат.

Кілька провідних постачальників ринку інтелектуальних транспортних систем (ITS), зокрема 3M Co., Advantech Co. Ltd., AECOM, Atlantia SPA, Axiomtek Co. Ltd., Cubic Corp., Dornier Group GmbH, EFKON GmbH, GMV Innovating Solutions SL, Hitachi Ltd., HNTB Co., HR Green Inc., Indra Sistemas SA, Kapsch TrafficCom AG, Serco Group Plc, SICE TyS, Siemens AG, Stantec Inc., Sumitomo Electric Industries Ltd., Teledyne FLIR LLC, Thales i Trapeze Software ULC .

Крім того є певна кількість систем комерційних та напів комерційних для аналізу та моделювання. Тобто це говорить як про важливість цього ринку так і про високу конкуренцію на ринку ITS. Таким чином для невеличкого старт ап краще вибрати стратегію фокусування на певних аспектах ITS ніж пробувати вирішити всі питання одночасно.

В таблиці 4.4 наведена характеристика потенційного ринку для розроблюваного стартап-проекту.

Таблиця 4.4

Попередня характеристика потенційного ринку стартап-проекту

№	Показники стану ринку	Характеристика
1	Кількість головних гравців, од	>20
2	Динаміка ринку (якісна оцінка)	Ринок зростає та стрімко рухається у напрямку використання AI технологій та розвитку систем V2I
3	Наявність обмежень для входу (вказати характер обмежень)	Вихід на великі ринки вкрай складний, але можливо реалізовувати певні рішення локально та фокусуючись на певних питаннях, сервісах.
4	Специфічні вимоги до стандартизації та сертифікації	Існують певні вимоги до систем, враховуючи критичну важливість роботи здатності систем.

Таблиця 4.5

Характеристика потенційних клієнтів стартап-проекту

№	Потреба, що формує ринок	Цільова аудиторія	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Реалізація концепції Смарт місто, боротьба з заторами, розвиток автономних транспортних засобів.	Адміністрації міст, транспортні компанії, сервісні компанії таксі сервіс, компанії розробники сигнальних систем.	В залежності від замовника питання коливається від мікро масштабу до макро масштабу.	Можливість модульного вирішення питань з отриманням певних результатів.

Таблиця 4.6

Фактори загроз

№	Фактор	Зміст загрози	Можлива реакція компанії
1	Висока конкуренція	Висока конкуренція обумовлена значним впливом транспортної системи на розвиток міст та економіки.	Компанія має сфокусуватись на певних аспектах реалізації та можливості інтеграції рішень у рішення конкурентів або використання їх сумісно.
2	Широкий спектр можливостей	Питання ITS покриває значну кількість можливосте, проте для невеликого старт ап треба вибрати потенційно сфокусоване рішення для більш ефективного його рішення та супроводження, особливо на локальному рівні	Можливий перегляд пріоритетних напрямків моделювання та застосування алгоритмів AI, але ITS генерую велику кількість даних для аналізу та можливості їх використання. Тому, є можливість гнучко реагувати на потреби ринку.
3	Економічний	Фінансові обмеження та висока конкуренція передбачають створення готових рішень на базі доступних технологій та використання у певних нішах, на індивідуальних або кластерних масивах.	Реалізація поступового просування моделі та демонстрація певних результатів.

Таблиця 4.7

Фактори можливостей

№	Фактор	Зміст можливості	Можлива реакція компанії
1	Пошук фінансування	Розробити певну модель на базі конкретних прикладів перехресть, наприклад у м. Київ та реалізувати підхід з використанням модуля управління світлофором	Потенційний позитивний результат дасть можливість масштабування проекту та переходу до більш складних шарів реалізації.
2	Розширення клієнтської бази	Головним чинником розширення проекту є його позитивна реалізація на пілотних проектах. Оскільки такого роду проекти на мають широкого кола споживачів то має працювати цільова робота з потенційними замовниками.	Спланувати пілотні проекти за власні кошти або кошти спонсорів. Надати результати реалізації. Розширити реалізацію пілотного проекту на більші масиви – кластери.
3	Заклучення контрактів на роботи	Реалізація пілотних проектів та проектів з спільним фінансуванням з основними замовниками – адміністраціями міст.	Сформувати потенційний перелік місць для тестування та реалізації проектів.

Таблиця 4.8

Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Складність входу на ринок	Критично важлива сфера, тому працюють багато компанії та підрядників, включаючи міжнародні корпорації та місцеві.	Вибрати стратегію фокусування на певних сервісах, та пропонувати різні схеми фінансування.
Загальнодоступність інформації	Інформація не є загальнодоступною та вимагає певного доступу. Крім того інформація має комерційний характер. Але є можливості як в плані збору даних так і в плані аналізу та застосування технологій.	Важливо розділити архітектуру на певні шари – для спрощення інтеграції та гнучкості у роботі з різними даними.
Схожість продуктів-замінників	Навіть якщо продукти замінники існують в даному випадку має значення локалізація, тому що умови роботи відрізняються.	Допомагати реалізовувати свої ідеї на місцях, тобто близько до замовника – це дає перевагу.

Таблиця 4.9

Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти в галузі	Постачальники	Клієнти	Товари-замінники
	3M Co., Advantech Co. Ltd., AECOM, Atlantia SPA, Axiomtek Co. Ltd., EFKON GmbH, GMV Innovating Solutions	Kapsch TrafficCom AG, Serco Group Plc, SICE TyS, Siemens AG, Stantec Inc., Sumitomo FLIR LLC, Thales	Постачальники в більшості це або мікроконтролери (Arduino, Raspberry та інші). Постачальники серверного обладнання та програмних продуктів.	Міська адміністрація, транспортні компанії, маркетингові компанії	VISSIUM, SUMO, та open source platforms та framework
Висновки:	Конкуренція висока	Програмних рішень достатньо, але є проблема локалізації та провадження на місцях	Постачальники диверсифіковані.	Перелік клієнтів в визначений та зрозумілий.	Основна перевага це команда для провадження на місцях.

Таблиця 4.10

Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Гнучка модель (модульна система реалізації)	Модель дозволяє впроваджувати елементи системи без ризику провалу.
2	Забезпечення крос платформенності	Модулі використовують Python та бібліотеки, які можуть працювати крос платформенно, але для систем які мають працювати у режимі реального часу варто використовувати C та C++.
3	Швидке реагування на відгуки	На етапі моделювання не приймалися до уваги існуючі технології програмування та хардвєа світлофорів, що може вимагати зміни формату даних наприклад.

Таблиця 4.11

Порівняльний аналіз сильних та слабких сторін

№	Фактор конкурентоспроможності	Бали 1-20	Порівняння з групами конкурентів		
			Група 1	Група 2	Група 3
1	Гнучка модель (модульна система реалізації)	15	9	12	15
2	Забезпечення кросплатформенності	10	15	12	10
3	Швидке реагування на відгуки	15	10	12	15

Таблиця 4.12

SWOT- аналіз стартап-проекту

Сильні сторони: Гнучкість модульної системи Відносна технічна нескладність простих рішень Швидке реагування на побажання клієнтів	Слабкі сторони: Значна конкуренція Недовіра великих клієнтів Індустріальні стандарти Невеликий досвід
Можливості: Реалізувати пілотні проекти за кошти донорів. Представити рішення конкретних проблем на місцевому рівні	Загрози: Програсти конкуренцію opensource проектам Невдала реалізація проекту, недостатньо доступу до даних первинних, для створення робочої моделі.

Таблиця 4.13

Альтернативи ринкового впровадження стартап-проекту

№	Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Надання моделі для тестування та тренування на реальних даних по місту Києву наприклад.	Можливість пошуку фінансування донорських проектів, особливо з боротьби з забрудненням середовища.	Якщо не враховувати отримання дозволів то локальна реалізація потребує принаймні збору даних протягом року. Аналіз та навчання моделі та реалізація не займає багато часу. Основний час це збір даних на конкретному перехресті.
2	Прийняття участі у донорських проектах, в тому числі і міжнародних.	Ймовірність висока	12 місяців
3	Участь в тендерах і укладення договорів з державними установами	Малоймовірно, оскільки програма має пройти сертифікацію та умовно допуск до таких систем.	Кілька місяців, в залежності від бюрократичних процедур

4.5. Розроблення ринкової стратегії

Таблиця 4.14

Вибір цільових груп потенційних споживачів

№	Опис профілю цільові групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Міністерство інфраструктури, Городські адміністрації, Автодор	Не готові	Продукт необхідний	Висока конкуренція	Складний вхід
2	Приватні компанії	Недостаток досвіду може обмежити прийняття	Певне обмежено коло споживачів	Конкуренція висока	Вхід помірно-складний
3	Навчальні інститути, школи, ентузіасти	Користувачі готові прийняти продукт	Не всі зацікавлені в продукті	Конкуренція середня	Вхід не є складний

Які цільові групи обрано:

Початковий план – робота з інститутами та навчальними закладами для можливості накопичення первинних даних для систем. Робота з іншими для запровадження практики збору даних для подальшого використання в роботі модулів програми.

Таблиця 4.15

Визначення базової стратегії розвитку

№	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Поетапний розвиток компанії та пошук донорських проектів.	Створення портфелю реалізованих проектів.	Залучення навчальних та донорських організацій до розповсюдження ідей пілотного використання	Модульний розвиток продукту, та фокусування на певних нішах сервісу та локальних ринках.

Таблиця 4.16

Визначення базової стратегії конкурентної поведінки

№	Чи є проект «пер- шопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	На даному етапі такої можливості немає, треба створювати власну нішу	Пропонується не копіювання, а створення можливостей інтеграції	Пропонувати рішення які можуть швидко надати результат який легко ідентифікується та позиціонується

Таблиця 4.17

Визначення стратегії позиціонування

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкуренто-спроможні позиції власного стартап- проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Надійність роботи. Наявність метрик ефективності (КПІ)	Доопрацювати модулі інтеграції з мікрочіпами або запропонувати модель використання мікрокомп'ютерів. Доопрацювати модуль аналізу мережі доріг та напрямку руху	Гнучкість до локальних вимог. Модульна система реалізації (мікросервісна)	Конструктор інтелектуальної системи управління транспортною мережею

4.6. Розроблення маркетингової програми проекту

Таблиця 4.18

Визначення ключових переваг потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Швидке впровадження ефективних рішень без суттєвих змін в інфраструктурній мережі доріг	Можливість із коробки мати інтеграцію з системою управління локальними світлофорами та застосування набору актуаторів для збору і аналізу інформації.	Фактично може продаватись як софт так і хард пакет з налаштуваннями. Мікросервісні модулі представляють архітектуру необхідну для провадження пошарово інтелектуальну транспортну систему.
2	Робото спроможність та стійкість системи	Можливість перепортування продукту на більш швидкі мови C++ та швидкість реагування	Створення відділу поліпшення та розвитку мікро сервісів в залежності від потреб замовників
3	Потреба мати чіткі та зрозумілі метрики та аналіз зібраних даних	Можливість проводити або додавати певні дані та аналізувати їх та готувати звіти для прийняття рішень.	Періодичні звіти щодо трафіку, аналіз трафіку, певні пропозиції.

Таблиця 4.19

Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові	
I. Товар за задумом	Конструктор, який складається з мікросервісних програм, які легко використовувати самостійно або інтегровано в основні існуючі системи ITS	
II. Товар у реальному виконанні	Властивості/характеристики	Розмір, без урахування OS
	Модуль-збору даних (актуаторів, камер, датчиків) та базового керування за принципом пріоритетності	70МБ (без урахування збереження отриманих даних)
	Модуль керування світлофорами	15МБ
	Модуль аналізу статистики та прогнозування трафіку	10МБ
	Якість: Програма потребує доопрацювання користувацького інтерфейсу.	
	Продаж послуг з збору даних, їх аналізу та імплементація системи управління.	
	Марка: OnTime	
III. Товар із підкріпленням	До продажу: послуги	
	Після продажу: послуги, імплементація і підтримка	
За рахунок чого потенційний товар буде захищено від копіювання: основна ідея створити послуги з збору та аналізу інформації, яка дозволить імплементувати програмний продукт та вести його обслуговування на базі існуючої інфраструктури транспортної системи		

Таблиця 4.20

Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі становлення ціни на товар/послугу
1				

*Ціновий аналіз не проводився стратегія мінімальних витрат тобто вартість менше конкурентів.

Таблиця 4.21

Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Покупка послуг, продаж даних та налаштувань	Надання інформації та налаштувань (навчаної мережі наприклад) онлайн	Прямий продаж, онлайн, підписникам, або замовникам	Право власності на отриманні дані може варіюватись в залежності від типу даних та умов.

Таблиця 4.22.

Концепція маркетингових комунікацій

№ п/ п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуютьс я цільові клієнти	Ключові позиції, обрані для позиціонуванн я	Завдання рекламного повідомленн я	Концепція рекламного звернення
1	Планують заздалегід ь	Тендерні закупівлі, пілотні проекти	Надання послуг, мікросервісів створення інтелектуально ї транспортної мережі	Донести необхідність хоча тестування можливостей системи	Легкість провадження , зрозумілість КП, очевидність результатів

Висновки до розділу 4

Четвертий розділ приділяє увагу розгляду потенційних можливостей продукту (сервісу) стати повноцінним продуктом та вийти на ринок. Бу проведений аналіз основних складових успішної стратегії та визначена позиція товару, основні конкуренти, слабкі та сильні сторони, модель позиціонування та інше. Для аналізу проекту було виконано:

1. Наведено опис сервісу та основні його складові. Проаналізовані ринкові умови та визначені особливості товару та ринку.
2. Проведено технічний аудит проекту та визначені його функціональні можливості, недоліки та необхідні напрямки розвитку.
3. Проаналізовано ринкові можливості вивести товар, запропонована певна стратегій фокусування на мікросервісах у загальній системі ITS.
4. Розроблено певну маркетингову стратегію з розуміння особливості ринку та високій конкуренції.

В підсумку, було отримано потенційну модель сервісного старт ап проекту для запуску мікросервісів у галузі інтелектуальної транспортної системи. Отримано певний досвід проведення аналізу продукту на життєдіяльність та потенційні слабкості та розробку стратегії реагування.

ВИСНОВКИ

Магістерська робота присвячена вирішенню проблем пов'язаних з поняттям інтелектуальної транспортної системи та окремих її елементів.

Було проведено аналіз матеріалів, щодо визначення теоретичних основ та практичного застосування понять транспортних потоків, існуючих систем регулювання та аналізу транспортних систем та методів їх аналізу та регулювання.

Також було розглянуто існуючі системи регулювання, як на законодавчому рівні та на рівні місцевому. Були розглянуті основні проблеми пов'язані з системою керування трафіком та попередження заторів. Основні акценти було зроблено на способах регулювання, прогнозування та управління транспортними потоками з метою покращення транспортних потоків та зменшення заторів. Що з іншого боку дає економічні вигоди у плані зменшення витрат та зменшення забруднення повітря.

Було обрано та аргументовано вибір набору алгоритмів для створення способу управління транспортними потоками, з метою запобігання заторам. Було розглянуто різні алгоритми обробки інформації, в залежності від рівня системи та необхідності реагування у реальному часі чи з можливістю певної затримки.

Для перевірки результатів роботи способу управління була розроблена модульна система (багатошарова) управління транспортними потоками, яка передбачає застосування певного переліку алгоритмів, у тому числі розпізнавання об'єктів, нелінійного аналізу статистики завантаження транспортних шляхів та інші.

Тестування роботи способу теоретично показало, що кожен з методів дає більш ефективний результат в плані економії часу транспортних засобів, таким чином доводячи дієвість методів активного регулювання порівняно з методами попереднього налаштування. Однак треба зауважити, що більшість цих тверджень базувались на дослідницьких роботах, та не перевірялись на практиці,

уданій магістерській роботі, у зв'язку з обмеженістю доступу до статистичної інформації м.Київ, щодо завантаженості транспортних шляхів у вільному доступі. Але було запропоновано вирішення цього питання шляхом використання середовищ симуляції, в даному випадку це була система SUMO. За певних умов налаштування в цій системі можна відтворити карту міста та симуляцію транспортних потоків, які можна використати для аналізу та моделювання алгоритмів аналізу та ефективності активного управління, як світлофорами так і індивідуальними авто.

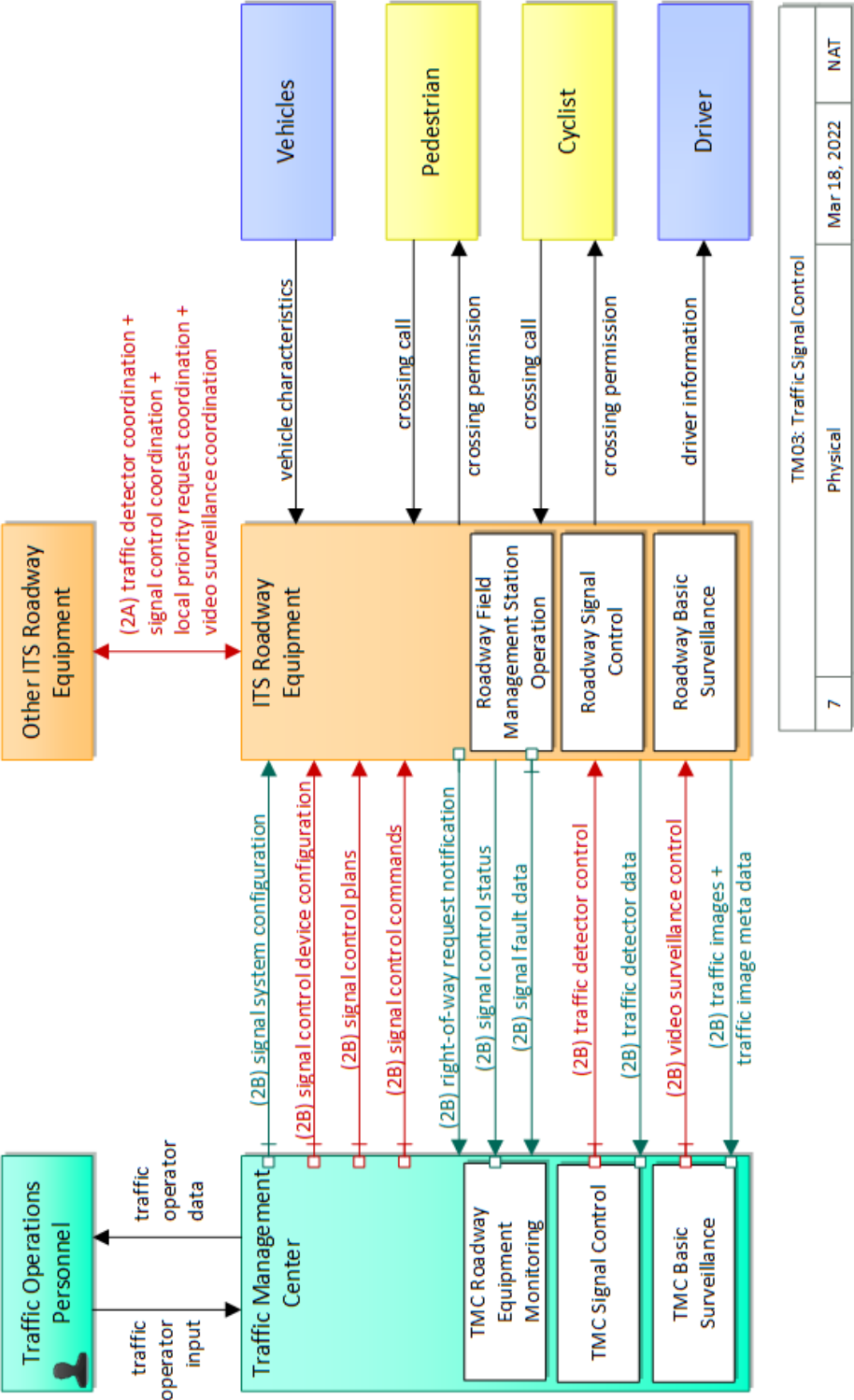
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. TRAFFIC CONTROL SYSTEMS HANDBOOK. [Електронний ресурс] // FEDERAL HIGHWAY ADMINISTRATION. – 2005. – Режим доступу до ресурсу: https://ops.fhwa.dot.gov/publications/fhwahop06006/fhwa_hop_06_006.pdf
2. Intelligent Transportation Systems Joint Program Office. [Електронний ресурс] // REGIONAL ITS ARCHITECTURE GUIDE - 2020 - Режим доступу до ресурсу: <https://www.arc-it.net/documents/raguide/raguide.pdf>
3. Systems Engineering for Intelligent Transportation Systems Handbook [Електронний ресурс] - Режим доступу до ресурсу: <https://www.arc-it.net/documents/seguide/seguide.pdf>
4. Intelligent Transportation Systems Joint Program Office. [Електронний ресурс] // Systems Engineering Tool for Intelligent Transportation (2022) - Режим доступу до ресурсу: <https://www.arc-it.net/tools/SET-ITHelpv91164.pdf>
5. AI enabled applications towards intelligent transportation. [Електронний ресурс] // School of Business and Management CHRIST, - Режим доступу до ресурсу: <https://doi.org/10.1016/j.treng.2021.100083>
6. R. Abduljabbar, H. Dia, S. Liyanage, S.A. Bagloee, [Електронний ресурс] // Applications of Artificial intelligence in transport: an overview, Sustainability 11 (189) - 2019 - Режим доступу до ресурсу: <https://doi.org/10.3390/su11010189>.
7. Intelligent Roadway Information System. [Електронний ресурс] - Режим доступу до ресурсу : https://mnit-rtmc.github.io/iris/road_topology.html
8. Intelligent Roadway Information System [Електронний ресурс] / Vitalii DutkaBusiness Director, Transportation & Mobility at Intellias - Режим доступу до ресурсу: <https://intellias.com/intelligent-traffic-management>
9. How to Solve Vehicle Routing Problems: Route Optimization Software and Their APIs [Електронний ресурс] - Режим доступу до ресурсу: <https://www.altexsoft.com/blog/business/how-to-solve-vehicle-routing-problems-route-optimization-software-and-their-apis>

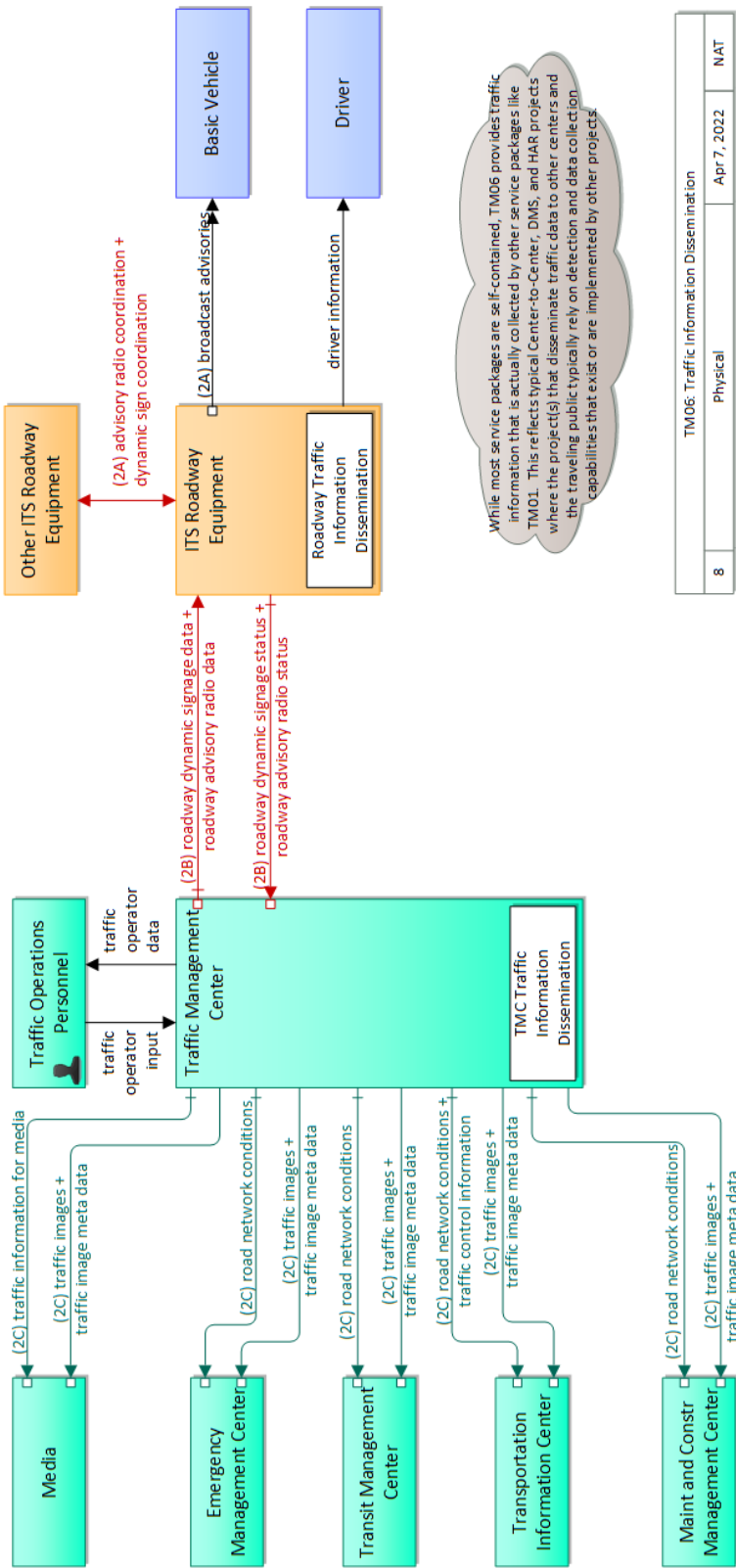
10. Traffic Prediction: How Machine Learning Helps Forecast Congestions and Plan Optimal Routes. [Электронный ресурс] - Режим доступа до ресурсу: <https://www.altexsoft.com/blog/traffic-prediction>
11. Vehicle Routing [Электронный ресурс] - Режим доступа до ресурсу: <https://developers.google.com/optimization/routing>
12. A Review of Traffic Congestion Prediction Using Artificial Intelligence/ Akhtar M, Moridpour S // [Электронный ресурс] – 2021. - Режим доступа до ресурсу: <https://doi.org/10.1155/2021/8878011>
13. “Microscopic traffic simulation using SUMO /P. L. Alvarez, et al. // International Conference on Intelligent Transportation Systems. IEEE (2018).
14. Design and Implementation of an Intelligent Traffic Management System: A Neural Approach / Shamim Akhter // [Электронный ресурс] - Режим доступа до ресурсу: <http://dx.doi.org/10.4018/978-1-5225-5832-3.ch004>
15. State-of-the-art of vehicular traffic flow modeling /Serge Hoogendoorn // [Электронный ресурс] - Режим доступа до ресурсу: <http://dx.doi.org/10.1243/0959651011541120>
16. On-Road Intelligent Vehicles Motion Planning for Intelligent Transportation Systems /Rahul Kala, // Robotics and Artificial Intelligence Laboratory, Indian Institute of Information Technology. – 2016.
17. Deep Learning on Traffic Prediction: Methods, Analysis and Future Directions /Xueyan Yin, Genze Wu, Jinze Wei, Yanming Shen, Heng Qi, and Baocai Yin
18. Dynamic spatio-temporal graph-based CNNs for traffic flow prediction. [Электронный ресурс] /Chen, K., Chen, F., Lai, B., Jin, Z., Liu, Y., Li, K., Wei, L., Wang, P., Tang, Y., Huang, J., Hua, X. // IEEE Access, 8, 185136–185145. – 2020. Режим доступа до ресурсу: <https://doi.org/10.1109/ACCESS.2020.3027375>
19. Traffic flow prediction models – A review of deep learning techniques. [Электронный ресурс] /Anirudh Ameya Kashyap, Shravan Raviraj, Ananya Devarakonda, Shamanth, R Nayak K, Santhosh K V & Soumya J Bhat - Режим доступа до ресурсу: <https://doi.org/10.1080/23311916.2021.2010510>

20. FRAME [Электронный ресурс] - Режим доступа до ресурсу:
<https://frame-online.eu/frame-architecture>
21. Incremental Machine Learning Platform for Big Data-Driven Smart Traffic Management [Электронный ресурс] / Dinithi Nallaperuma, Rashmika Nawaratne, Tharindu Bandaragoda , Achini Adikari, Su Nguyen, Member // IEEE, Thimal Kempitiya, Daswin De Silva, Member, IEEE, Damminda Alahakoon, Member, IEEE, and Dakshan Pothuhera, - Режим доступа до ресурсу:
https://www.researchgate.net/publication/334410680_Online_Incremental_Machine_Learning_Platform_for_Big_Data-Driven_Smart_Traffic_Management.
22. ARC-IT [Электронный ресурс] - Режим доступа до ресурсу:
<https://www.arc-it.net/html/servicepackages/>
23. Traffic Signal Control System Based on Intelligent Transportation System and Reinforcement Learning. Electronics [Электронный ресурс] / Hurtado-Gómez, J.; Romo, J.D.; Salazar-Cabrera, R.; Pachón de la Cruz, Á.; Madrid Molina, J.M. 2021, 10, 2363. - Режим доступа до ресурсу: <https://doi.org/10.3390/electronics10192363>
24. Real-time object detection YOLOv3 [Электронный ресурс] / Karlijn Alderliesten, article - Режим доступа до ресурсу: <https://medium.com/analytics-vidhya/yolov3-real-time-object-detection-54e69037b6d0>
25. Colight [Электронный ресурс]/ Hua Wei*, Nan Xu*, Huichu Zhang, Guanjie Zheng, Xinshi Zang, Chacha Chen, Weinan Zhang, Yanmin Zhu, Kai Xu, Zhenhui Li - Режим доступа до ресурсу:
<https://sites.psu.edu/huawei/2019/09/15/colight-cikm-2019>
26. Y. Lv, Y. Duan, W. Kang, Z. Li and F. Wang, "Traffic Flow Prediction With Big Data: A Deep Learning Approach", IEEE Trans. Intell. Transp. Syst., vol. 16, no. 2, pp. 865-873, Apr. 2015.
27. R. Fu, Z. Zhang and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), 2016, pp. 324-328, doi: 10.1109/YAC.2016.7804912.

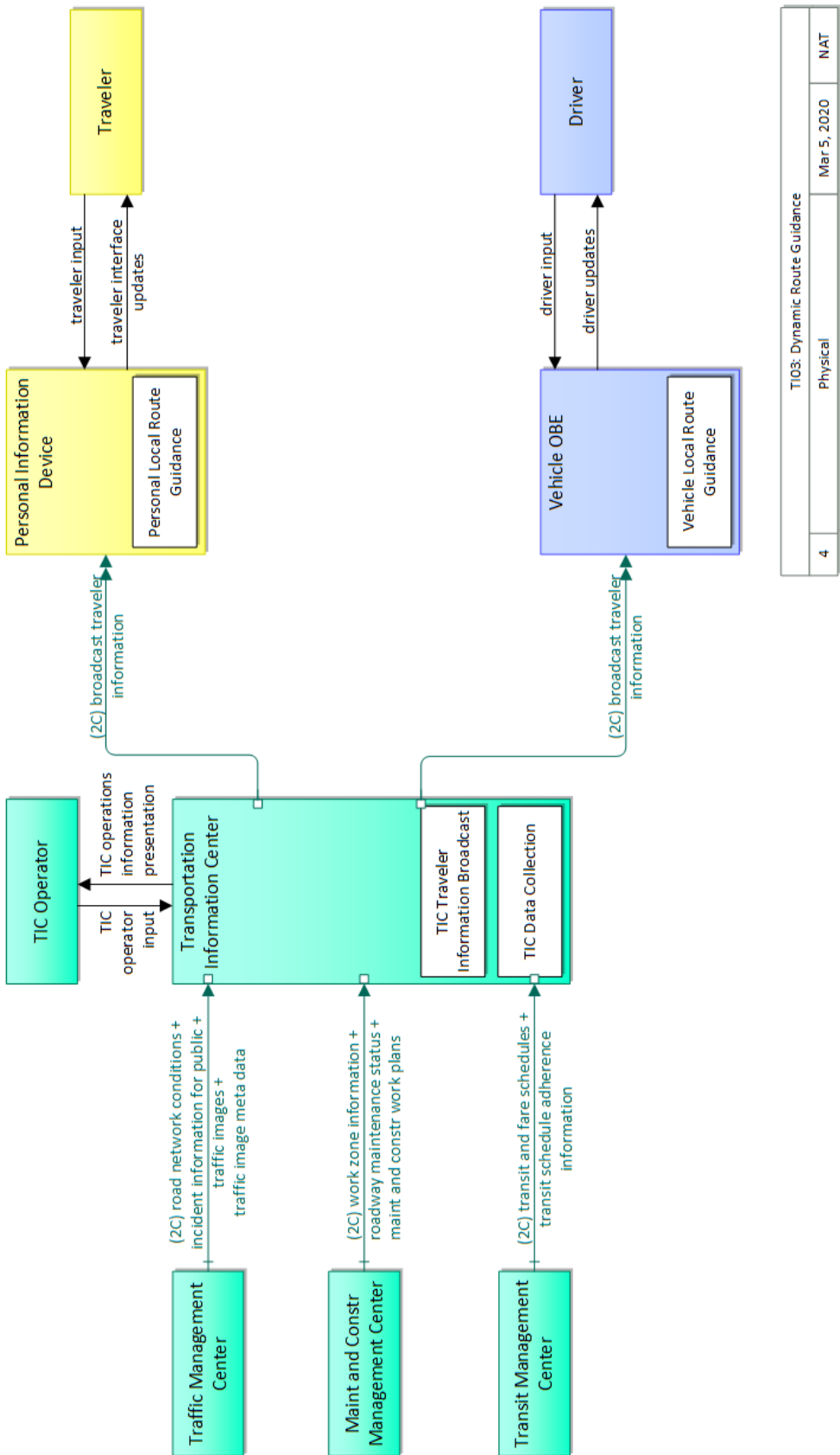
Контроль системи світлофорів (ARC-IT) [22]



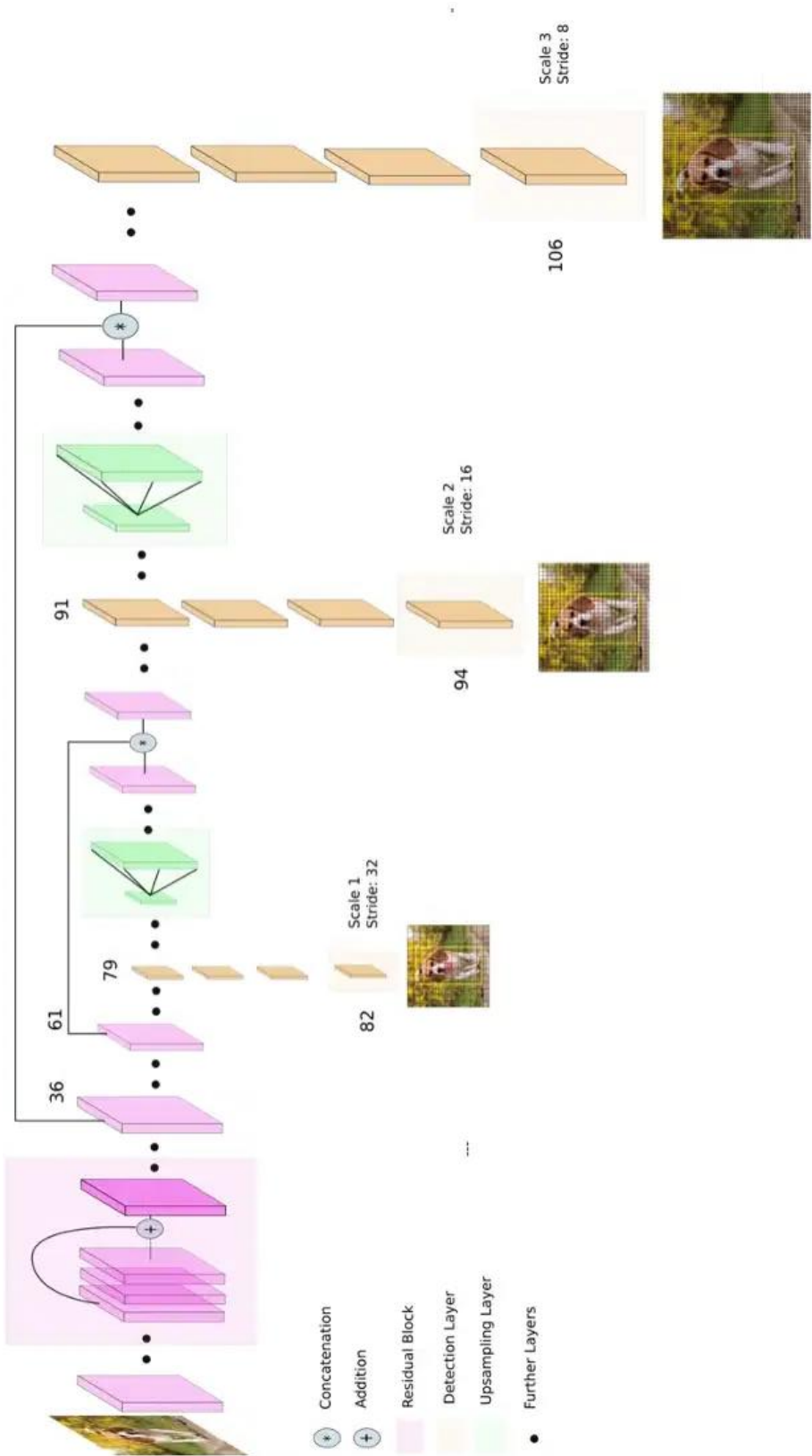
Система розповсюдження інформації (ARC-IT)[22]



Динамічна система побудови маршруту (ARC-IT)[22]



Архітектура YOLO нейромережі [24]



Псевдокод DQN алгоритма

Algorithm 1 DQN algorithm

```

1: Input Replay memory size  $M$ , batch size  $d$ , number of
   episodes  $E$ , and number of time steps  $T$ 
2: Initialize Main network weights  $\theta$ 
3: Initialize Target network weights  $\theta^-$ 
4: Initialize Replay memory
5: for  $e = 1, \dots, E$  do
6:   Initialize state  $s_1$ , and action  $a_1$ 
7:   for  $t = 1, \dots, T$  do
8:     Take action  $a_t = \operatorname{argmax}_a Q^\pi(s_t, a; \theta)$  with prob-
       ability  $1 - \epsilon$  or a random action with probability  $\epsilon$ 

9:     Get reward  $r_t$  and observe next state  $s_{t+1}$ 
10:    if Replay capacity  $M$  is full then
11:      Delete the oldest tuple in memory
12:    end if
13:    Store the tuple  $(s_t, a_t, r_t, s_{t+1})$  to replay memory
14:    Sample random  $d$  tuples from replay memory
15:     $y_t = \begin{cases} r_t, & \text{if } t = T. \\ r_t + \gamma \max_a Q^\pi(s_{t+1}, a_{t+1}; \theta_t^-), & \text{otherwise.} \end{cases}$ 
16:    Perform policy gradient using  $y_t$  for updating  $\theta$ 
17:    Update target network every  $N$  step,  $\theta^- = \theta$ 
18:  end for
19: end for

```

Псевдокод Q алгоритма

Algorithm 1: Q-Learning

Result: Off-policy control for estimating $\pi \simeq \pi_*$
 Parameters: step size $\alpha \in (0, 1]$, small $\epsilon > 0$
 Initialize $Q(s, a) \forall s \in \mathcal{S}^+, a \in \mathcal{A}(s)$ arbitrarily.
 Set $Q(\text{terminal}, \cdot) = 0$
for each episode **do**
 Initialize S ;
 for $\text{step} = 0$ **to** T **do**
 Choose A from S using policy derived from Q (e.g., ϵ -greedy);
 Take action A , observe R, S' ;
 $Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma \max_a Q(S', a) - Q(S, A)]$;
 $S \leftarrow S'$;
 end
end

Псевдокод Random Forest алгоритма

Algorithm 1 Random Forest

Precondition: A training set $S := (x_1, y_1), \dots, (x_n, y_n)$, features F , and number of trees in forest B .

- 1 **function** RANDOMFOREST(S, F)
- 2 $H \leftarrow \emptyset$
- 3 **for** $i \in 1, \dots, B$ **do**
- 4 $S^{(i)} \leftarrow$ A bootstrap sample from S
- 5 $h_i \leftarrow$ RANDOMIZEDTREELEARN($S^{(i)}, F$)
- 6 $H \leftarrow H \cup \{h_i\}$
- 7 **end for**
- 8 **return** H
- 9 **end function**
- 10 **function** RANDOMIZEDTREELEARN(S, F)
- 11 At each node:
- 12 $f \leftarrow$ very small subset of F
- 13 Split on best feature in f
- 14 **return** The learned tree
- 15 **end function**

Псевдокод SVR алгоритма

Data : Dataset with p^* variables and binary outcome.

Output: Ranked list of variables according to their relevance.

Find the optimal values for the tuning parameters of the SVM model;

Train the SVM model;

$p \leftarrow p^*$;

while $p \geq 2$ **do**

$SVM_p \leftarrow$ SVM with the optimized tuning parameters for the p variables and observations in **Data**;

$w_p \leftarrow$ calculate weight vector of the SVM_p ($w_{p1}, \dots, w_{pp}$);

$rank.criteria \leftarrow (w_{p1}^2, \dots, w_{pp}^2)$;

$min.rank.criteria \leftarrow$ variable with lowest value in $rank.criteria$ vector;

 Remove $min.rank.criteria$ from **Data**;

$Rank_p \leftarrow min.rank.criteria$;

$p \leftarrow p - 1$;

end

$Rank_1 \leftarrow$ variable in **Data** $\notin (Rank_2, \dots, Rank_{p^*})$;

return ($Rank_1, \dots, Rank_{p^*}$)

Псевдокод SAEs алгоритма

Algorithm 1 Pseudocode for the training algorithm of the Sparse-Autoencoder.

Load database: $\mathbf{z}^{\text{GT}}$ (train partition);
Define network architecture: $N_{\text{in}}^{\text{SAE}} = N_{\text{out}}^{\text{SAE}} = D, N_h^{\text{SAE}}, N_d^{\text{SAE}}, \sigma_j^{\text{SAE}};$
Define hyperparameters: $\lambda_r^{\text{SAE}}, \lambda_j^{\text{SAE}};$
Initialize $w_{i,j}^{\text{SAE}}, b_j^{\text{SAE}};$
for each epoch **do**
 Initialize loss function: $C = 0;$
 for each train snapshot **do**
 Encoder: $\mathbf{x}_n^{\text{SAE}} = q_\phi(\mathbf{z}_n^{\text{GT}});$
 Decoder: $\mathbf{z}_n^{\text{SAE}} = p_\theta(\mathbf{x}_n^{\text{SAE}});$
 Loss function: $C \leftarrow C + \mathcal{L}_n^{\text{rec}} + \lambda_r^{\text{SAE}} \mathcal{L}_n^{\text{reg}};$
 end for
MSE loss function: $\mathcal{L}^{\text{SAE}} \leftarrow \frac{C}{N_{\text{train}}}$
Backward propagation;
Optimizer step;
end for

Фрагмент програмного коду**Реалізація DQN алгоритму**

```
import random
import torch
import torch.nn as nn
import nnarch
from collections import namedtuple
import copy
import math

device = "cuda" if torch.cuda.is_available() else "cpu"

Transition = namedtuple('Transition',
                        ('state', 'action', 'next_state', 'reward'))

class DqnAgent:
    def __init__(
        self,
        mode: str,
        replay,
        target_update: int,
        gamma: float,
        use_sgd: bool,
        eps_start: float,
        eps_end: float,
        eps_decay: int,
        input_dim: int,
```

```

    output_dim: int,
    batch_size: int,
    network_file: str = "
):
    self.mode = mode
    self.replay = replay
    self.target_update = target_update
    self.gamma = gamma
    self.use_sgd = use_sgd
    self.eps_start = eps_start
    self.eps_end = eps_end
    self.eps_decay = eps_decay
    self.n_actions = output_dim
    self.batch_size = batch_size

    self.network_file = network_file
    self.policy_net = nnarch.DqnNetwork(input_dim, output_dim).to(device)
    self.target_net = nnarch.DqnNetwork(input_dim, output_dim).to(device)
    self.policy_net_copy = nnarch.DqnNetwork(input_dim, output_dim).to(device)
    if network_file:
        self.policy_net.load_state_dict(torch.load(network_file,
map_location=torch.device(device)))
        self.target_net.load_state_dict(self.policy_net.state_dict())
        self.target_net.eval()

    self.learn_steps = 0
    self.z = None
    self.fixed_gamma = copy.deepcopy(gamma)
    self.update_gamma = False
    self.q_value_batch_avg = 0

```

```

def select_action(self, state, steps_done, invalid_action):
    original_state = state
    state = torch.from_numpy(state)
    if self.mode == 'train':
        sample = random.random()
        eps_threshold = self.eps_end + (self.eps_start - self.eps_end) * math.exp(-1. *
steps_done / self.eps_decay)
        if sample > eps_threshold:
            with torch.no_grad():
                _, sorted_indices = torch.sort(self.policy_net(state), descending=True)
                if invalid_action:
                    return sorted_indices[1]
                else:
                    return sorted_indices[0]
        else:
            decrease_state = [(original_state[0] + original_state[4]) / 2,
                               (original_state[1] + original_state[5]) / 2,
                               (original_state[2] + original_state[6]) / 2,
                               (original_state[3] + original_state[7]) / 2]
            congest_phase = [i for i, s in enumerate(decrease_state) if abs(s-1) < 1e-2]
            if len(congest_phase) > 0 and invalid_action is False:
                return random.choice(congest_phase)
            else:
                return random.randrange(self.n_actions)
    else:
        with torch.no_grad():
            _, sorted_indices = torch.sort(self.policy_net(state), descending=True)
            if invalid_action:
                return sorted_indices[1]

```

```

        else:
            return sorted_indices[0]

def learn(self):
    if self.mode == 'train':
        if self.replay.steps_done <= 10000:
            return
        loss_fn = nn.MSELoss()
        if self.use_sgd:
            optimizer = torch.optim.SGD(self.policy_net.parameters(), lr=0.00025)
        else:
            optimizer = torch.optim.RMSprop(self.policy_net.parameters(), lr=0.00025)

        transitions = self.replay.sample(self.batch_size)
        batch = Transition(*zip(*transitions))
        state_batch = torch.cat(batch.state)
        action_batch = torch.cat(batch.action).view(self.batch_size, 1)
        next_state_batch = torch.cat(batch.next_state)
        reward_batch = torch.cat(batch.reward).view(self.batch_size, 1)
        state_action_values = self.policy_net(state_batch).gather(1, action_batch)
        with torch.no_grad():
            argmax_action = self.policy_net(next_state_batch).max(1)[1].view(self.batch_size, 1)
            q_max = self.target_net(next_state_batch).gather(1, argmax_action)
            expected_state_action_values = reward_batch + self.gamma * q_max
            # for plot
            self.q_value_batch_avg = torch.mean(state_action_values).item()

        loss = loss_fn(state_action_values, expected_state_action_values)
        optimizer.zero_grad()

```

```

loss.backward()

self.cal_z(state_batch, action_batch, q_max)

for param in self.policy_net.parameters():
    param.grad.data.clamp_(-1, 1)
optimizer.step()
self.learn_steps += 1
self.update_gamma = False

def cal_z(self, state_batch, action_batch, q_max):
    self.policy_net_copy.load_state_dict(self.policy_net.state_dict())
    z_optimizer = torch.optim.SGD(self.policy_net_copy.parameters(), lr=0.0001)
    state_action_copy_values = self.policy_net_copy(state_batch).gather(1,
action_batch)
    z_optimizer.zero_grad()
    f_gamma_grad = torch.mean(0.00025 * q_max * state_action_copy_values)
    f_gamma_grad.backward()
    self.z = {'l1.weight': self.policy_net_copy.l1.weight.grad,
              'l1.bias': self.policy_net_copy.l1.bias.grad,
              'l2.weight': self.policy_net_copy.l2.weight.grad,
              'l2.bias': self.policy_net_copy.l2.bias.grad}

def learn_gamma(self):
    loss_fn = nn.MSELoss()
    optimizer = torch.optim.SGD(self.policy_net.parameters(), lr=0.00025)

    transitions = self.replay.sample(self.batch_size)
    batch = Transition(*zip(*transitions))
    state_batch = torch.cat(batch.state)

```

```

action_batch = torch.cat(batch.action).view(self.batch_size, 1)
next_state_batch = torch.cat(batch.next_state)
reward_batch = torch.cat(batch.reward).view(self.batch_size, 1)
state_action_values = self.policy_net(state_batch).gather(1, action_batch)
with torch.no_grad():
    argmax_action =
self.policy_net(next_state_batch).max(1)[1].view(self.batch_size, 1)
    q_max = self.target_net(next_state_batch).gather(1, argmax_action)
    expected_state_action_values = reward_batch + self.fixed_gamma * q_max

loss = loss_fn(state_action_values, expected_state_action_values)
optimizer.zero_grad()
loss.backward()

l1_weight = self.policy_net.l1.weight.grad * self.z['l1.weight']
l1_bias = self.policy_net.l1.bias.grad * self.z['l1.bias']
l2_weight = self.policy_net.l2.weight.grad * self.z['l2.weight']
l2_bias = self.policy_net.l2.bias.grad * self.z['l2.bias']

gamma_grad = -0.99 * torch.mean(torch.cat((l1_weight.view(-1), l1_bias.view(-
1), l2_weight.view(-1), l2_bias.view(-1))))
self.gamma += gamma_grad
self.update_gamma = True

```

Архітектура мережі DQN

```

import torch
import torch.nn as nn
import torch.nn.functional as F
import numpy as np

device = "cuda" if torch.cuda.is_available() else "cpu"

def normalize_output(outputs):
    with torch.no_grad():
        for m, output in enumerate(outputs):
            max_t = torch.max(torch.abs(output))
            if abs(max_t-0) < 1e-2:
                continue
            for n, t in enumerate(output):
                outputs[m][n] /= max_t
    return outputs

class DqnNetwork(nn.Module):
    def __init__(self, inputs, outputs):
        super(DqnNetwork, self).__init__()
        self.l1 = nn.Linear(inputs, 512)
        self.l2 = nn.Linear(512, outputs)
        c = np.sqrt(1 / inputs)
        nn.init.uniform_(self.l1.weight, -c, c)
        nn.init.uniform_(self.l1.bias, -c, c)
        nn.init.uniform_(self.l2.weight, -c, c)
        nn.init.uniform_(self.l2.bias, -c, c)

```

```
def forward(self, x):
    x = x.to(device)
    return x
```

"Прогноз дорожнього руху за мережами LSTM, SAE, GRU"[26, 27]

```
import math
import warnings
import numpy as np
import pandas as pd
import tensorflow as tf
from data.data import process_data
from keras.models import load_model
from keras.utils.vis_utils import plot_model
import sklearn.metrics as metrics
import matplotlib as mpl
import matplotlib.pyplot as plt
warnings.filterwarnings("ignore")
```

```
tf.compat.v1.disable_v2_behavior()
```

```
def MAPE(y_true, y_pred):
    y = [x for x in y_true if x > 0]
    y_pred = [y_pred[i] for i in range(len(y_true)) if y_true[i] > 0]

    num = len(y_pred)
    sums = 0
```



```

for i in range(num):
    tmp = abs(y[i] - y_pred[i]) / y[i]
    sums += tmp

```

```

mape = sums * (100 / num)

```

```

return mape

```

```

def eva_regress(y_true, y_pred):
    mape = MAPE(y_true, y_pred)
    vs = metrics.explained_variance_score(y_true, y_pred)
    mae = metrics.mean_absolute_error(y_true, y_pred)
    mse = metrics.mean_squared_error(y_true, y_pred)
    r2 = metrics.r2_score(y_true, y_pred)
    print('explained_variance_score:%f' % vs)
    print('MAPE:%f%%' % mape)
    print('MAE:%f' % mae)
    print('MSE:%f' % mse)
    print('RMSE:%f' % math.sqrt(mse))
    print('R2:%f' % r2)

```

```

def plot_results(y_true, y_preds, names):
    d = '2022-3-4 00:00'
    x = pd.date_range(d, periods=288, freq='5min')

    fig = plt.figure()
    ax = fig.add_subplot(111)

```

```
ax.plot(x, y_true, label='Дані з доріг')
for name, y_pred in zip(names, y_preds):
    ax.plot(x, y_pred, label=name)
```

```
plt.legend()
plt.grid(True)
plt.xlabel('Час дня')
plt.ylabel('Потік')
```

```
date_format = mpl.dates.DateFormatter("%H:%M")
ax.xaxis.set_major_formatter(date_format)
fig.autofmt_xdate()
```

```
plt.show()
```

```
def main():
```

```
    lstm = load_model('model/lstm.h5')
    gru = load_model('model/gru.h5')
    saes = load_model('model/saes.h5')
    models = [lstm, gru, saes]
    names = ['LSTM', 'GRU', 'SAEs']
```

```
    lag = 12
```

```
    file1 = 'data/train.csv'
```

```
    file2 = 'data/test.csv'
```

```
    _, _, X_test, y_test, scaler = process_data(file1, file2, lag)
```

```
    y_test = scaler.inverse_transform(y_test.reshape(-1, 1)).reshape(1, -1)[0]
```

```

y_preds = []
for name, model in zip(names, models):
    if name == 'SAEs':
        X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1]))
    else:
        X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
    file = 'images/' + name + '.png'
    plot_model(model, to_file=file, show_shapes=True)
    predicted = model.predict(X_test)
    predicted = scaler.inverse_transform(predicted.reshape(-1, 1)).reshape(1, -1)[0]
    y_preds.append(predicted[:288])
    print(name)
    eva_regress(y_test, predicted)

plot_results(y_test[: 288], y_preds, names)

if __name__ == '__main__':
    main()

```