

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)
(рівень вищої освіти)

Система моніторингу та управління транспортними маршрутами на основі
графових структур
(тема)

Виконав: здобувач IV курсу, групи КІУКІ-21-8
Голубовський Н. О.
(прізвище, ініціали)


Спеціальність 123 Комп'ютерна інженерія
Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія
(повна назва освітньої програми)

Керівник Чумаченко С.В
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри АПОТ


Чумаченко С.В
(підпис) (прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління

Кафедра Автоматизації проектування обчислювальної техніки

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 Комп'ютерна інженерія
(шифр і назва)

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

Освітня програма Комп'ютерна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри АПОТ



Чумаченко С.В.

(підпис)

« 05 » 05 2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Голубовський Никита Олексійович
(прізвище, ім'я, по батькові)

1. Тема роботи (проекту) Система моніторингу та управління транспортними маршрутами на основі графових структур

затверджена наказом по університету від "21" 05 2025 р. № 403 Ст.

2. Термін подання студентом роботи до екзаменаційної комісії 15.06.2025

3. Вихідні дані до роботи (проекту)

Моделі та методи для пошуку оптимальних шляхів на графових структурах

Топології графових структур

4. Зміст пояснювальної записки (перелік питань, що підлягають розробці):

Огляд технологій

Аналіз моделей та інфраструктури керування транспортом

Розробка структури системи моніторингу та управління транспортними маршрутами на основі графових топологій

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 20 слайдів

6. Консультанти розділів роботи (проекту)

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

КАЛЕНДАРНИЙ ПЛАН


№	Назва етапів роботи (проекту)	Термін виконання етапів проекту (роботи)	Примітка
1	Видача теми проекту, узгодження і затвердження	06.05.2025 -06.05.2025	
2	Аналіз проблемної галузі, постановка задачі, вибір інструментальних засобів. Складання аналітичного огляду стану технологій. Аналіз останніх досліджень та публікацій. Написання вступу та огляду моделей й алгоритмів.	07.05.2025 -10.05.2025	
3	Опис обраних моделей, методів, алгоритмів, критеріїв, інфраструктури	10.05.2025 -17.05.2025	
4	Розробка структури системи моніторингу та управління транспортними маршрутами на основі графових топологій	18.05.2025 -23.05.2025	
5	Опис сервісів системи	24.05.2025 -30.05.2025	
6	Оцінка алгоритму	01.06.2025 -05.06.2025	
7	Оформлення пояснювальної записки	06.06.2025 -10.06.2025	
8	Перевірка виконаного проекту керівником, допуск до захисту	11.06.2025 -12.06.2025	
9	Захист проекту	26.06.2025	

Дата видачі завдання 06.05.2025

Здобувач _____  Голубовський Н.О.

(підпис)

Керівник роботи
(підпис)


(посада, прізвище, ініціали)

проф.Чумаченко С.В.

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 54 с., 13 рисунків, 4 табл., 1 дод., 35 джерел.

СИСТЕМА, ГРАФ, КРИТЕРІЙ, ТОПОЛОГІЯ, МЕТРИКА, МОДЕЛЬ, ТЕСТУВАННЯ, ПОШУК, СТРУКТУРА, ХМАРНИЙ СЕРВІС, МАРШТУР, АЛГОРИТМ

Робота пов'язана з підвищенням ефективності управління дорожнім рухом шляхом розробки системи моніторингу та управління транспортними маршрутами на основі графових структур з її подальшою інтеграцією у хмарний сервіс для online обчислення оптимального маршруту автомобіля.

В роботі розглянуто стан технологій; виконано аналіз публікацій, моделей та e-інфраструктури дорожнього руху; розробка структури системи моніторингу та управління транспортними маршрутами на основі графових топологій з онлайн обчисленням оптимального маршруту автомобіля, що включає модифікований алгоритм Дейкстри для трикутної та прямокутної топології.

ABSTRACT

Bachelor's thesis contains 54 pages format A4, 13 figures, 4 tables, 1 application, 35 sources.

SYSTEM, GRAPH, CRITERION, TOPOLOGY, METRIC, MODEL, TESTING, SEARCH, STRUCTURE, CLOUD SERVICE, ROUTE, ALGORITHM

The work is related to improving the efficiency of traffic management by developing a system for monitoring and managing transport routes based on graph structures, with its subsequent integration into a cloud service for online calculation of the optimal route for a car.

The thesis considers the state of technology; analyzes publications, model models and e-infrastructure of road traffic; development of a structure for monitoring and management of transport routes based on graph topologies with online calculation of the optimal route of the car, including a modified Dijkstra algorithm for triangular and rectangular topology.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	9
1 СТАН ТЕХНОЛОГІЙ	10
1.1 Технологічні тенденції в автомобільній електроніці	10
1.2 Аналіз літературних джерел	14
1.3 Висновки до розділу 1	18
2 МОДЕЛІ, МЕТРИКА ТА ІНФРАСТРУКТУРА	19
2.1 Хмарні обчислення та системи керування транспортом	19
2.2 Вимірювання відстаней на E-інфраструктурі дорожнього руху	25
2.3 Структура зв'язків компонентів системи	30
2.4 Якість топології комп'ютерної системи	34
2.5 In-метогукомп'ютинг автономного транспорту	37
2.6 Висновки до розділу 2	37
3 СИСТЕМА МОНІТОРИНГУ ТА УПРАВЛІННЯ ТРАНСПОРТНИМИ МАРШРУТАМИ НА ОСНОВІ ГРАФОВИХ СТРУКТУР	39
3.1 Модифікація алгоритму Дейкстри для окремих топологій	39
3.2 Сервіси хмарного керування транспортом	45
3.3 Висновки до розділу 3	48
ВИСНОВКИ	49
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	50
ДОДАТОК А Графічна частина	55

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ

ADAS – AdvancedDriver-AssistanceSystems (удосконалена система допомоги водію);

AI – Artificial Intelligence (штучний інтелект);

ASIC –Application-SpecificIntegratedCircuit (інтегральна схема для конкретного застосування);

BMS – BatteryManagementSystem (система керування батареєю);

CCD – Charge-CoupledDevice (прилад із зарядовим зв'язком);

CMOS – ComplementaryMetal-Oxide-Semiconductor (комплементарний метал-оксид-напівпровідник, КМОП);

DSRC – DedicatedShortRangeCommunications (спеціалізований зв'язок ближнього радіусу дії);

ECU – ElectronicControlUnit (електронний блок керування);

ESFRI – EuropeanStrategyForumonResearchInfrastructures(Європейський стратегічний форум з дослідницьких інфраструктур);

FANSCI – FastHandoverSchemeforCar-to-CarCommunication (схема швидкої передачі зв'язку між автомобілями);

FCC – FederalCommunicationsCommission(Федеральна комісія по зв'язку у США);

IEEE – InstituteofElectricalandElectronicsEngineers (Інститут інженерів з електротехніки та електроніки);

LTE– ультрамобільний;

MCU – MicrocontrollerUnit;

MIHF – MediaIndependentHandoverFunction (функція незалежної передачі даних від медіа);

MIES – сервіс медіа-незалежних подій;

MICS – сервіс медіа-незалежних команд;

MIS – сервіс медіа-незалежної інформації;

MOR – MyOwnRange (мій власний асортимент);

MOSFET – Metal-Oxide-SemiconductorField-EffectTransistor (польовий транзистор метал-оксид-напівпровідник);

SoC – System-on-Chip (система на кристалі);

WLAN – WirelessLocalAreaNetwork (безпроводова локальна мережа);

АЦП– аналого-цифровий перетворювач;

ВЧ – висока частота;

ПЛІС – програмована логічна інтегральна схема.

ВСТУП

Тематика роботи стосується питань створення системи моніторингу та управління транспортними маршрутами на основі графових структур як компонента хмарного сервісу Smart Cloud Traffic Control.

Мета дослідження – підвищення ефективності управління дорожнім рухом шляхом розробки системи моніторингу та управління транспортними маршрутами на основі графових структур з її подальшою інтеграцією у хмарний сервіс для online обчислення оптимального маршруту автомобіля.

Задачі: огляд стану технологій; аналіз публікацій, аналіз моделей та e-інфраструктури дорожнього руху; розробка системи моніторингу та управління транспортними маршрутами на основі графових структур з online обчисленням оптимального маршруту автомобіля, що включає модифікований алгоритм Дейкстри для трикутної та прямокутної топології.

1 СТАН ТЕХНОЛОГІЙ

1.1 Технологічні тенденції в автомобільній електроніці

Впровадження цифрових онлайн-сервісів хмарного інтелектуального трафіку тягне за собою позитивні соціальні, технологічні та технічні трансформації. Інноваційний хмарний онлайн-сервіс SmartCloudTrafficControl [1] пропонує впровадження часового параметра у цифрову карту планети, перенесення світлофорів і дорожніх знаків у віртуальний кіберпростір. Це завершує формування віртуальної інтелектуальної дорожньої інфраструктури, спрямованої на підвищення якості життя водіїв, скорочення часу в дорозі, зменшення витрат на паливо, а також заощадження тисяч тон металу та мільйонів коштів, необхідних для виготовлення, встановлення й обслуговування світлофорів і знаків. Крім цього, сервіс дозволяє економити мільйони кіловат-годин електроенергії для їх утримання та зводить час на оновлення дорожньої інфраструктури у віртуальному просторі міст до кількох хвилин. Усі ці переваги сприяють автоматизації процесів квазіоптимального управління транспортом і трафіком у реальному часі, а також вирішенню соціальних, гуманітарних, економічних і екологічних завдань.

Тренди автомобільної електроніки[2]. Друковані плати є основою автомобільної електроніки, на яку в даний час припадає ~40% вартості нового автомобіля. Очікується, що до 2030 року цей показник досягне 50%, так само як і нові автомобілі будуть частково або повністю автономними.

Справа не тільки в кількості електронних компонентів, яка, як очікується, продовжить зростати, але і в складності цих систем, як з точки зору апаратного забезпечення, так і з точки зору прошивки. За цією статистикою стоїть кілька тенденцій в автомобільній електроніці, які визначатимуть потреби в компонентах.

Основні тенденції в автомобільній електроніці. Автомобільна електроніка охоплює різноманітні сфери застосування, від джерела живлення до датчиків і бездротового зв'язку. Найпомітнішою тенденцією є нинішній дефіцит автомобільних чіпів. На жаль, вона поширилася практично на всі інші сфери електронної промисловості.

Управління енергоспоживанням для електромобілів. У міру зміни інфраструктури в економічно розвинених країнах і розробки нових акумуляторних систем можна очікувати електромобілі з більшим запасом ходу і швидшою зарядкою. Все це залежить від більш досконалих систем управління енергією, які покладаються на ряд основних компонентів. Ці компоненти не обов'язково мають високу інтеграцію, як SoC, оскільки їм потрібно керувати великою кількістю енергії, але системи з високою потужністю все одно можуть вимагати роботи з дискретними компонентами на спеціалізованих модулях. Деякі з основних систем живлення, що з'являються в електромобілях, включають: бездротові системи BMS для управління розподілом заряду в акумуляторах електромобілів, а також для моніторингу акумуляторів і передачі даних назад на блоки управління; поява технологій V2G та двонаправленої зарядки в електромобілях на зарядних станціях; використання більш досконалих напівпровідників з високою робочою температурою і високою теплопровідністю для силових MOSFET.

Оскільки електромобілі працюють з акумуляторними блоками при високій напрузі, компоненти захисту від електростатичного розряду також важливі для захисту ланцюгів. Використовуються інтегровані мікросхеми управління живленням з безліччю регуляторів, а також автомобільні процесори та набір ASIC. Процесор додатків NXPMC33PF8200A0ES забезпечує інтегроване рішення для управління живленням для автомобільної промисловості.

Мережна взаємодія в автомобілі та розумна інфраструктура. Нові авто мобілі обробляють більше даних, ніж будь-коли раніше, і обсяг даних, які вони використовують, буде тільки збільшуватися. Наразі швидкість

підключення до мережі в споживчих автомобілях становить менше 1 Гбіт/с через Ethernet, але гігабітний Ethernet в автомобілі та бездротові пристрої всередині автомобіля змінять спосіб збору та обробки даних, а також досвід водія. Мережна взаємодія в автомобілі та інтелектуальна інфраструктура є величезними можливостями для нових автомобілів і розглядаються як новий зростаючий ринок для автомобільної промисловості. За прогнозами, до 2026 року ринок лише мережної взаємодії в автомобілі становитиме 1,5 мільярддоларів, що сприятиме використанню різноманітних інтегрованих процесорів і SoC. CC2541-Q1 від Texas Instruments має автомобільну кваліфікацію та є частиною платформи SimpliLink для продуктівIoT. Мережева проводка в сучасному автомобілі можевключати в себе тисячі підключень і розтягуватися на тисячі метрів в довжину. Заміна дротових інтерфейсів на бездротові з'єднання зменшує вагу та складність системи, і це відповідає поточній парадигмі, яка зустрічається в підключених автомобілях.

Щоб забезпечити більш просунуті ADAS, автономні транспортні засоби та низку нових послуг, що надходять зсередини та ззовні автомобіля, нові автомобілі також повинні підключатися один до одного за допомогою інтелектуальних систем інфраструктури, а також з велосипедами та мотоциклами.

Поточні стандарти бездротового зв'язку «від автомобіля до всього» (V2X) базуються на WLAN та IEEE 802.11p, тоді як інші можливості залежатимуть від наявних 4G або майбутніх стільникових служб 5G. Компоненти, необхідні для цих систем, включають наступне: 4G/4G LTE та 5G трансивери, модеми та антенні модулі; мікросхеми підсилювача потужності ВЧ для каналів передачі; ВЧ трансиверні мікросхеми, що працюють в діапазоні частот (WiFi, DSRC, дводіпазонні системи до 5,9 ГГц). До недавнього часу точилися дебати про те, чи стане стандарт 802.11p (відомий як спеціалізований короткостроковий зв'язок, або DSRC) або стільниковий зв'язок домінуючим протоколом для міжтранспортної мережі. У жовтні 2020 року FCC перерозподілила спектр від 5,85 ГГц до 5,895 ГГц у

несанкціонованому діапазоні. Решта оригінального спектру DSRC була виділена для новішого C-V2X, який щойно стандартизований у 3GPP Release 14. Це фактично поклато край DSRC, залишивши послуги на основі C-V2X та 5G як майбутні технології для підключених автомобілів та розумної інфраструктури.

Автомобіль може виступати як точка доступу Wi-Fi і підключатися до стільникового зв'язку через 4G LTE/5G (рекламується як послуги підключеного автомобіля), а потім підключатися до пристроїв через Bluetooth. У міру розвитку 5G можливості стільникового зв'язку продовжуватимуть відображатися на нових автомобілях. Що стосується нових автомобілів, ми повинні очікувати продовження переходу на новіші бездротові технології, щоб забезпечити передовий користувацький досвід.

Датчики для інтелектуальних систем ADAS. Експерти очікують, що приблизно через десятиліття споживачі зможуть придбати безпілотний автомобіль. Причин тому безліч, які в основному пов'язані з розробкою набору передових алгоритмів контролю і прийняття рішень. Однак є й інші проблеми, які потрібно вирішувати на апаратному рівні. Крім того, виникають регуляторні проблеми та інфраструктура, необхідна для підтримки автономних транспортних засобів.

Поточне різноманіття датчиків для ADAS включає певну комбінацію ультразвукових датчиків, радарів і камер, усі з яких повинні бути підключені до ECU. Лідар також може стати звичайним явищем у мережах датчиків ADAS, оскільки він дозволяє створювати карти глибини, що неможливо із зображеннями з камер. Лідар цікавий тим, що його використання не обмежується автомобілями, технологія корисна для виявлення та візуалізації в розумних містах взагалі. Хоча лідар не був основною темою для обговорення в останні роки, компанії все ще просувають його як частину передового рішення для інтелектуальних систем ADAS, оскільки він забезпечує зображення та картографування з високою роздільною здатністю для підтримки радарних та візуальних систем у нових автомобілях.

Компоненти, необхідні для датчиків ADAS, виходять за рамки самих датчиків і включають в себе наступне: радіопередавачі та SoC (24 ГГц та 77 ГГц); компоненти системи Lidar, в тому числі імпульсні лазерні діоди; багатоканальні АЦП і вбудовані процесори для об'єднання даних з датчиків; компоненти для кастомних камер, включаючи сенсорні модулі CCD/CMOS.

Обчислювальна потужність. Сучасні автомобілі містять понад 100 ECU, і очікується, що це число буде тільки зростати. У міру збільшення обсягу даних, що збираються і обробляються в автомобілях, ECU та інші модулі системи будуть вимагати більше обчислювальної потужності у вигляді мікроконтролерів і автомобільних ПЛІС. Точний розмір, швидкість і розташування цих компонентів поки що залишаються відкритими питаннями.

З огляду на тенденцію до інтеграції, спостерігаючи на спеціалізованому ринку мікросхем, можна очікувати, що багато виробників почнуть пропонувати та/або рекламувати SoC для автомобілів, які інтегрують MCU. Неможливо йти в ногу з усіма тенденціями в автомобільній електроніці, але виробники чіпів, швидше за все, відповідатимуть спеціалізованими SoC, подібними до тих, що використовуються для IoT та мобільних продуктів.

1.2 Аналіз літературних джерел

Кіберфізична система онлайн-хмарного управління транспортом спрямована на забезпечення сучасних сервісів як для водіїв, так і для транспортних компаній. На сьогодні аналогічних комплексних систем на глобальному рівні не існує, хоча окремі компоненти, що сприяють створенню відповідної інфраструктури, вже розроблені та використовуються. Серед них можна виокремити електронні карти, супутникові системи визначення місцеположення та навігації, спеціалізовані бази даних у хмарному середовищі, технології моніторингу кіберзахисту, комп'ютери для автівок, гаджети з приймачами для водіїв, світлофори з централізованим керуванням, а також системи мобільного зв'язку.

Фінансова доступність оренди для водіїв мобільних хмарних сервісів навігації, управління та моніторингу руху транспортних засобів в межах 100 доларів на рік. Наявність програмних, апаратних та мережевих систем централізованого управління дорожнім рухом у масштабах країни. Доступність хмарних обчислювальних технологій, постійне їх вдосконалення та здешевлення для створення інфраструктури доріг у кіберпросторі. Зростання комп'ютерної, мобільної та інтернет грамотності населення.

Тематика досліджень ґрунтується на публікаціях:

1. Розробка теоретичних основ інтелектуальних моделей, методів і програмно-апаратних засобів для аналізу кіберпростору, пов'язаних із дискретною оптимізацією, пошуком, розпізнаванням і прийняттям рішень [3, 4].
2. Напрацьований досвід у створенні й використанні вбудованих та RFID цифрових систем для моніторингу дорожнього руху [5-8].
3. Реалізація розробки та впровадження програмних рішень і хмарних сервісів, спрямованих на оптимізацію маршрутів транспортних засобів для українських підприємств, що дозволяє мінімізувати матеріальні та часові витрати, а також підвищувати якість обслуговування пасажирів [9-13].
4. Розробка розподілених систем управління дорожнім рухом для великих міст і мегаполісів, що базуються на високонадійній обчислювальній техніці [10–11], а також прототипування сучасних кіберфізичних систем керування трафіком [16–17].
5. Методи оцінки обчислювальних структур і пошуку найкоротших шляхів між парами вершин, а також відомі модифікації алгоритму Дейкстри [16–18].
6. Чинні системи моніторингу дорожнього руху, такі як OnStar у США та Канаді, а також NEXCO Central у Японії. Наразі сервіс обслуговує приблизно 4 мільйони користувачів. Вартість одного пристрою моніторингу становить близько 200 доларів. Дані від сенсорів, серед яких переважно датчики ударів і спрацювання подушок безпеки, автоматично передаються до

call-центрів, що дає змогу оперативно інформувати рятувальні й правоохоронні служби про дорожню ситуацію.

7. Трансконтинентальні автомобільні магістралі, підводні тунелі, вражаючі мости, а також космічні й авіаційні проєкти виступають потужними драйверами економічного розвитку багатьох країн. Паралельно стрімкий розвиток Інтернету, мобільних технологій та хмарних обчислень відкрив можливість для безмежного збільшення обчислювальних потужностей і обсягів зберігання даних. Наступний етап у розвитку кіберпростору – це Інтернет речей, що передбачає взаємодію пристроїв між собою без участі людини, автоматичний збір, обробку й аналіз великих даних, які створюються за допомогою розумних сенсорів.

8. Концепція Інтернету речей (IoT) вимагає значних інновацій в комп'ютерній інженерії. Розробка та реалізація IoT привертає увагу провідних наукових організацій, таких як IEEE та EAI, а також великих технологічних компаній, включаючи Intel, NASA, IBM, Apple, Google, Samsung, Dell та інші. Концептуальна основа Інтернету речей базується на використанні бездротових гетерогенних мереж, які функціонують у різних частотних діапазонах відповідно до таких стандартів, як ZigBee, WiFi, LTE (4G, 5G) та Bluetooth. Ці мережі забезпечують необхідний баланс між дальністю дії, швидкістю передачі даних і низьким рівнем енергоспоживання. Ключовою функціональною складовою інфраструктури IoT є процес збору і аналізу даних, який реалізується з використанням технологій великих даних (bigdata), відкритих даних (opendata) та хмарних обчислень (cloudcomputing). Одним із фундаментальних елементів IoT виступає міжмашинна взаємодія (Machine-to-Machine, M2M) – комплекс технологій і підходів, що забезпечують автоматизований обмін інформацією між пристроями. Це створює передумови для розбудови розумних міст. До таких рішень долучаються компанії, які пропонують платформи та готові апаратні розробки, зокрема Connit, Aguila Technologies та WorldSensing. Одним із прикладів успішних проєктів є FastPrk від компанії WorldSensing —

це сервіс для організації «розумного паркування», що сприяє полегшенню пошуку вільних місць для водіїв за допомогою мобільних пристроїв. Водночас міська адміністрація отримує можливості для більш ефективного управління паркувальними зонами. Сенсори цього сервісу працюють у частотному діапазоні Sub-GHz з радіусом дії до 500 метрів.

9. Сучасний автомобіль важко уявити без інтеграції інтернет-сервісів, які поєднують комфортну поїздку, зручний відпочинок та можливість здійснення бізнес-операцій безпосередньо в салоні автомобіля. Останній виконує роль універсального простору з повним спектром функціональних послуг для ділової людини, таких як голосовий зв'язок, обмін повідомленнями, електронна пошта, банкінг, навігаційні системи, а також розваги у вигляді музики, відео та ігор. Інтеграція автомобіля в мережу Інтернет, що досягається шляхом з'єднання мобільного пристрою з бортовою системою, забезпечує ідентифікацію водія в реальному та цифровому просторах. Найбільш помітними учасниками цієї технологічної сфери є компанії Apple та Google. Вони представили платформи інтегрованих комунікаційних, навігаційних та розважальних систем (Apple iOS CarPlay та Google Android Auto), функціонування яких базується на використанні мікропроцесорної операційної системи BlackBerry QNX.

10. Оновлена структура всесвітньої інженерної спільноти IEEE з її секціями, такими як IEEE Geoscience та Remote Sensing Society, IEEE Intelligent Transportation Systems Society та інші, підкреслює ринкову та науково-технічну перспективність проекту SmartCloudTrafficControl. Ці секції IEEE, що охоплюють сфери геопросторових даних, інтелектуальних транспортних систем, хмарних обчислень та інших, демонструють актуальність та затребуваність розробок, подібних до SmartCloudTrafficControl, у сучасному інженерному світі.

1.3 Висновки до розділу 1

Проаналізовано технологічні тенденції щодо автомобільної електроніки. Проведено аналіз сучасних досліджень та публікацій. Робиться висновок про те, що впровадження цифрових онлайн-сервісів хмарного інтелектуального трафіку тягне за собою позитивні соціальні, технологічні та технічні трансформації, дозволяє автоматично керувати транспортними потоками та транспортом майже оптимальним чином в реальному часі.

Мета роботи – підвищення ефективності управління дорожнім рухом шляхом розробки системи моніторингу та управління транспортними маршрутами на основі графових структур з її подальшою інтеграцією у хмарний сервіс для online обчислення оптимального маршруту автомобіля.

Задачі: огляд стану технологій; аналіз публікацій, аналіз моделей та е-інфраструктури дорожнього руху; розробка системи моніторингу та управління транспортними маршрутами на основі графових структур з online обчисленням оптимального маршруту автомобіля, що включає модифікований алгоритм Дейкстри для трикутної та прямокутної топології.

2 МОДЕЛІ, МЕТРИКАТА ІНФРАСТРУКТУРА

Аналізуються основні моделі, метрики та критерії для аналізу ефективності транспортної інфраструктури, е-інфраструктура дорожнього руху, складові кіберфізичної системи для відстеження та регулювання трафіку в режимі реального часу.

2.1 Хмарні обчислення та системи керування транспортом

У сучасному світі швидкими темпами розвивається електронна інфраструктура (Е-інфраструктура) дорожнього руху, що активно впроваджується в межах концепції Internet of Car.

У роботах [19-23] висвітлюється інформація щодо використання як існуючої, так і перспективної Е-інфраструктури для підвищення якості життя населення. Розвиток електронної інфраструктури (Е-інфраструктури) є критично важливим для сучасних наукових досліджень та багатьох інших сфер. Від організації онлайн-конференцій з використанням хмарних технологій до забезпечення роботи державних служб, "розумних" міст та управління транспортом у реальному часі – все це залежить від надійної Е-інфраструктури. Важливість цього напрямку підтверджується діяльністю ESFRI – європейської організації, яка координує зусилля університетів та компаній для створення стандартів та прототипів Е-інфраструктур, що забезпечують ефективну роботу кіберфізичних систем. Сучасна Е-інфраструктура в Європі включає в себе мережу дата-центрів, хмарні сервіси, системи аутентифікації та інтерфейси для зв'язку з хмарами. Останні десять років показали, що попит на Е-інфраструктуру та хмарні сервіси, які забезпечують цілодобовий онлайн-доступ до послуг, залишається стабільно високим.

Хмарні сервіси [24, 25] надають зручний та масштабований інтерфейс для доступу до додатків та ресурсів, повністю керованих провайдером. Масштабування є критично важливим для задоволення змінних потреб

користувачів, одночасно мінімізуючи витрати на розгортання та підтримку локальної інфраструктури. В умовах поширеного мобільного доступу та високої пропускної здатності мереж, користувачі можуть ефективно співпрацювати незалежно від географічного розташування. Архітектура системи повинна включати чітко визначені політики контролю доступу до даних та сервісів для учасників з різних державних та приватних доменів. Управління ідентифікацією та довірою є ключовим шаром у цифровій інфраструктурі екосистеми. Федеративні протоколи аутентифікації та авторизації спрощують доступ до міжорганізаційних ресурсів, забезпечуючи безпеку при роботі з гетерогенними джерелами даних. Це дозволяє користувачам отримувати надійне середовище для зберігання важливих і цілісних даних без ризику їх компрометації. Вказаний компонент інтегрує обчислювальні можливості мереж із інфраструктурою даних, створюючи відкритий, безпечний і надійний європейський цифровий простір.

У [26] представлено стратегічне бачення розвитку британської електронної інфраструктури, яка розглядається як об'єднання політичних, технологічних та адміністративних послуг. Її основна мета – забезпечити зручний і економічно доцільний доступ до розподілених електронних ресурсів для їх спільного використання. Електронну інфраструктуру слід сприймати як гнучку екосистему, яка залишається відкритою для міжнародної співпраці і здатна адаптуватися до швидко змінюваних умов суспільного життя. Прогрес у створенні окремих компонентів цієї інфраструктури спрямований на те, щоб спростити виконання складних бізнес-обчислень, дозволяючи науковцям зосередитися на своїх дослідженнях. Варто підкреслити, що успішний розвиток цієї галузі залежить від наявності відкритої конкуренції, співпраці та технологічних інновацій. Реалізація електронної інфраструктури можлива лише за умови розробки спільного стратегічного плану, в якому братимуть участь як користувачі, так і організації, відповідальні за її створення та функціонування. Практика демонструє, що здатність генерувати, збирати й аналізувати великі обсяги

даних для застосування у сфері промисловості, науки й державного управління стає вагомим фактором підвищення конкурентоспроможності та економічного зростання кожної країни.

Електронна інфраструктура має ключове значення для стимулювання інновацій у багатьох секторах. Наприклад, вона сприяє більш ефективному використанню можливостей та економії ресурсів, які стають доступними завдяки технологічному прогресу [27]. Вона також є основою для виконання великомасштабних процесів моделювання, аналізу і візуалізації даних у межах екосистеми ресурсів. Така інфраструктура забезпечує розподілене співробітництво, обчислення та розробку інструментів для максимально ефективного використання наявних даних і ресурсів (рис. 2.1).

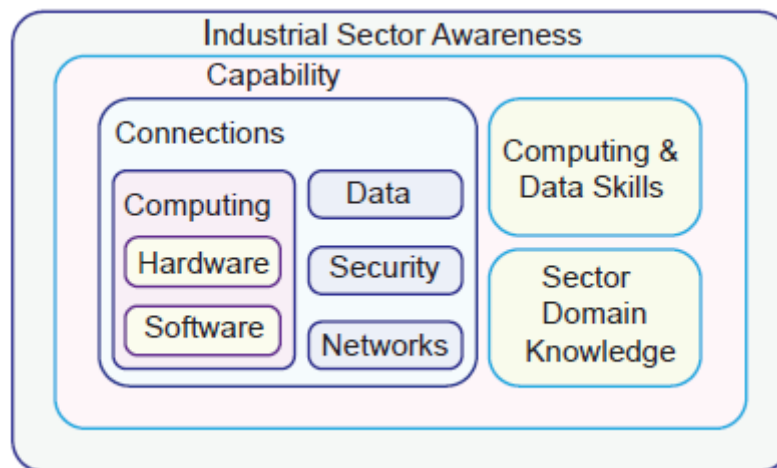


Рисунок 2.1 – Схема компонентів Е-інфраструктури[33]

У праці [28], опублікованій ще у 2002 році, було досліджено еволюцію систем керування міським транспортом, зокрема, застосування комп'ютерних технологій для регулювання трафіку. Автори показали, як розвиток обчислювальної техніки, локальних мереж та хмарних технологій сприяв вдосконаленню методів управління, що трансформувалися від централізованих моделей до розподілених систем, які використовують мобільні пристрої та хмарні сервіси. Запропоновано систему паралельного управління транспортом (PtMS), ключовим елементом якої є "Штучна

транспортна система". Ця система передбачає використання віртуального середовища, що працює паралельно з реальною транспортною мережею. Віртуальне середовище містить бази даних зі стандартними шаблонами управління трафіком, агентами та сценаріями руху. Будь-які датчики, що збирають дані про трафік, використовуються як мобільні сенсори. Основна ідея полягає в інтелектуальній обробці даних, отриманих від дорожніх сенсорів, за допомогою хмарних технологій. Рішення щодо управління трафіком приймаються на основі заздалегідь визначених алгоритмів і стратегій. У складних ситуаціях, коли відсутні готові шаблони, завдання розбивається на простіші підзадачі. Запропонований у статті підхід "local simple, remote complex" спрямований на інтелектуальну організацію управління транспортом у рамках концепції Smart Cloud Traffic Control. PtMS є частиною системи "Adapts" (Adaptive Platforms for Transportation Systems), яка складається з трьох рівнів: організації, координації та виконання завдань з обробки дорожніх ситуацій. Організаційний рівень включає чотири функції: декомпозицію задачі, планування, інкапсуляцію та управління. Це дозволяє значно підвищити якість регуляторних впливів завдяки використанню складніших алгоритмів. Однак, система має обмеження: за певний час вона може обробити лише обмежену кількість перехресть. Для запуску та тестування системи на сервері потрібні значні обчислювальні ресурси та обсяг пам'яті. Ці обмеження не відповідають реальним умовам дорожнього руху у великих містах, тому для управління трафіком на великій кількості перехресть в режимі реального часу необхідно використовувати декілька серверів. Для зберігання даних про трафік та дорожні ситуації потрібні великі центри обробки даних. У статті запропоновано два шляхи вирішення цієї проблеми: використання суперкомп'ютерів або застосування хмарних обчислювальних технологій, таких як Google's Map-Reduce, IBM Blue Cloud та Amazon.

У документі [29] розглянуто стандарт IEEE 802.21, який забезпечує підтримку різних типів бездротових мереж, зокрема сімейства протоколів

IEEE 802.11, IEEE 802.16 (WiMAX) і стільникових технологій передачі даних, розроблених групами 3GPP і 3GPP2. Стандарт IEEE 802.21 реалізовує функцію Media Independent Handover (MIH), що забезпечує незалежну передачу даних у стільникових мережах. Основу фреймворку IEEE 802.21 формують два ключові компоненти: Media Independent Handover Function (MIHF) та точки доступу до сервісів (Service Access Points, SAP). Функція MIHF є модуль, що надає сервіси через медіа-незалежний інтерфейс для вищих рівнів моделі передачі даних. MIHF включає три основні сервіси MIES / MICS / MINS – медіа-незалежних подій/ команд /інформації . Стандартизовані точки доступу до сервісів і примітиви забезпечують уніфікований доступ додатків MIHF до різноманітних рівнів мережевих медіа. У рамках розвитку цього стандарту запропоновано систему FANSCI (FastHandoverSchemeForCar-to-CarCommunication), що адаптує IEEE 802.21 для передачі даних, необхідних для забезпечення постійного та безперебійного зв'язку в гетерогенних мережах. Система FANSCI містить два основні елементи: базову станцію, розташовану у центрі управління мережею, та автомобільний компонент, інтегрований у транспортний засіб.

Автомобільний компонент містить кілька важливих модулів для забезпечення комунікації та обробки даних. Зокрема, до його складу входить модуль Car-to-Car, який об'єднує менеджер з'єднання, збору GPS-даних, сенсори автомобіля та інформацію про двигун. Окрім цього, на стороні клієнта розташований модуль обробки даних, який виконує такі функції: встановлює з'єднання (менеджер ініціації), відстежує сигнали, зберігає дані у буфері та передає їх. Модуль зв'язку між автомобілями (Car-to-Car), що включає обробку трафіку, перевірку автентичності та управління маршрутизацією, а також модуль передачі даних, який відповідає за обробку запитів, взаємодію з MIH-функціями, буферизацію трафіку та перенаправлення даних. Технологія FANSCI може бути використана в будь-яких бездротових мережах завдяки основі в стандартах IEEE 802.21, підкріпленому IEEE 802.11 (Wi-Fi) та IEEE 802.16 (WiMAX). Водії за

допомогою цієї технології отримують можливість постійно відстежувати інформацію про дорожні умови: пробки, перехрестя, несправні транспортні засоби, а також доступні гетерогенні мережі. Такий підхід стає основою для створення безпілотних автомобілів.

Дослідження [30] показують, що хмарне керування здатне допомагати уникати небезпечних ситуацій під час водіння. Автономні транспортні засоби, призначені для міських умов, можуть обробляти інформацію і приймати рішення відповідно до дорожніх знаків та правил. Вони здатні виконувати маневри типу «стоп-старт», перемикання смуги руху, обгін або вихід з потоку транспорту. Такі ТЗ оснащені радарми, GPS-навігацією, інерційними системами та іншими дорогими компонентами. Запропонований новий алгоритм управління в хмарі має високу продуктивність і гнучкість. Для створення ефективної автономної системи водіння вирішуються три ключові завдання: самостійне керування автомобілем (старт, уповільнення, і повороти); підтримка керування уздовж дороги чи по центру смуги; забезпечення безпеки на дорозі для водіїв і пішоходів шляхом визначення перешкод. Програмна архітектура складається з таких компонентів: інтерфейс сенсорів дороги, система збору інформації про перешкоди і смуги, маршрутизатор та інтерфейс ТЗ. Апаратна архітектура включає `SensorLayer`, `PreprocessLayer`, `IntegrationLayer` і `VehicleLayer`. Для уникнення зіткнень автономні автомобілі повинні розпізнавати ситуацію як на своїй смузі руху, так і на сусідніх смугах, подібно до сприйняття водіїв у реальних умовах. Хмарний алгоритм управління використовує метод MOR (`MyOwnRange`) для оцінки кута і відстані між автомобілем та перешкодами через ультразвукові датчики і камери. У складних ситуаціях, наприклад при зміні смуги сусіднім транспортним засобом з увімкненим сигналом повороту, алгоритм проводить прогнозування на основі дистанції й кута повороту. Майбутній рух автомобіля також можна оцінити за стилем водіння: нормальне водіння, уповільнення і гальмування, обгін з лівої або правої смуги. Для цього система генерує оптимальні команди щодо прискорення, гальмування та

управління кермом за допомогою спеціалізованих контролерів, дотримуючись дорожніх правил.

2.2 Вимірювання відстаней на E-інфраструктурі дорожнього руху

Для успішної ідентифікації та розпізнавання елементів, об'єктів, процесів або подій у кіберпросторі потрібна дієва система вимірювання відстаней у дискретному булевому просторі (просторі логічних векторів). Ця система має бути основою для швидких алгоритмів, які визначають ступінь подібності або відмінності між цими елементами [31, 32].

Кіберпростір, представлений як дискретний векторно-логічний простір, є сукупністю інформаційних процесів і явищ, які взаємодіють між собою згідно з певною метрикою. Ці процеси та явища описуються векторами логічних змінних і використовують пам'ять як носій інформації.

Метрика визначає спосіб обчислення відстані між компонентами процесів або явищ, представлених векторами логічних змінних. У кіберпросторі відстань визначається як XOR-відношення між двома векторами, що описують компоненти процесу або явища, і відрізняється від кодової відстані Хеммінга. Поняття відстані, похідної (булевої), ступеня відмінності або схожості є взаємозамінними і використовуються для визначення відношення між двома компонентами процесу чи явища. У кіберпросторі, відстань між компонентами кількісно визначає їхню несхожість. Процеси порівняння, вимірювання, оцінювання, розпізнавання, тестування, діагностування та ідентифікації слугують для визначення взаємозв'язків між об'єктами, за умови наявності щонайменше двох.

Компонент простору поданок-вимірним (двійковим) вектором

$$a = (a_1, \dots, a_j, \dots, a_k), a_j \in \{0, 1\}, k = 1, 2, 3, \dots, \quad (2.1)$$

де кожний його компонент 0 – «неправда», 1 – «істина». Нульовий вектор ϵ_k -вимірною упорядкованою послідовністю, всі координати якого дорівнюють нулю: $a_j = 0, j = \overline{1, k}$.

Метрика b кіберпростору задається співвідношенням, що формує нульовий вектор як додавання за модулем два відстаней d_i між ненульовою та кінцевою кількістю об'єктів, які є замкнутими к циклу:

$$\beta = \bigoplus_{i=1}^n d_i = 0, \quad (2.2)$$

де n – це число, що визначає кількість відстаней між компонентами або векторами простору, $D = (d_1, \dots, d_i, \dots, d_n)$ є вектор відстані зі складовими циклу d_i , що відповідають ребру циклу, яке з'єднує пару компонентів a, b простору, що далі позначається без індексу як $d(a, b)$. Відстань між двома об'єктами (векторами) a і b є вектор: $d(a, b) = (a_j \oplus b_j)_1^k$. Векторному значенню відстані відповідає норма, що представляє собою скалярну відстань за Хемінгом між двома векторами, дорівнює кількості одиничних елементів у векторі $d(a, b)$, тобто метрика b векторного логічного двійкового простору є дорівнена нульовому вектору сума за модулем два відстаней між кінцевим числом вершин графа, які замкнені у цикл. Сума n -вимірних двійкових векторів, що задають координати точок циклічної фігури, дорівнює нульовому вектору.

На основі введеної метрики кіберпростір можна розглядати більш формально, а саме: як векторно-логічним, нормованим за b -метрикою, де сума за модулем два відстаней між кінцевим числом точок циклу дорівнює нульовому вектору. Метрика визначається не через властивості окремих елементів, а через відношення між ними, що дає змогу звести три базові аксіоми (тотожність, симетрію та трикутну нерівність) до однієї. Ця спрощена аксіоматика може бути застосована до будь-яких, навіть дуже складних, конструкцій у n -вимірному логічному просторі. Визначення

взаємодії між точками (одна, дві або три) у векторному просторі є лише частковим випадком застосування метрики b -метрики при $i = 1, 2, 3$:

$$M = \begin{cases} d_1 = 0 \leftrightarrow a = b; \\ d_1 \oplus d_2 = 0 \leftrightarrow d(a, b) = d(b, a); \\ d_1 \oplus d_2 \oplus d_3 = 0 \leftrightarrow d(a, b) \oplus d(b, c) = d(a, c). \end{cases} \quad (2.3)$$

Векторно-логічний транзитивний трикутник можна розглядати як аналог числового вимірювання відстані в метричному просторі M . Ключовою особливістю цього простору є система аксіом, яка визначає, як відстані між точками (одна, дві або три) пов'язані між собою:

$$M = \begin{cases} d(a, b) = 0 \leftrightarrow a = b; \\ d(a, b) = d(b, a); \\ d(a, b) + d(b, c) \geq d(a, c). \end{cases} \quad (2.4)$$

У метричному просторі, аксіома трикутника, хоч і встановлює зв'язок між відстанями трьох точок, не дає змоги точно визначити довжину однієї сторони трикутника, знаючи довжини двох інших. Невизначеність полягає в тому, що сума двох сторін лише обмежує можливий діапазон довжини третьої. На відміну від цього, підхід, заснований на векторній логіці, дозволяє уникнути цієї невизначеності. Він забезпечує детерміноване визначення бінарних відношень, що описують стан процесу або явища. Замість діапазону можливих значень для третьої сторони, векторна логіка дає точний двійковий вектор, який однозначно визначає відстань між двома точками, використовуючи інформацію про відстані двох інших сторін трикутника:

$$d(a, b) \oplus d(b, c) = d(a, c) \rightarrow d(a, b) \oplus d(b, c) \oplus d(a, c) = 0 \quad (2.5)$$

У багатозначному векторно-логічному кіберпросторі метрикою b є вектор, що є результатом застосування операції симетричної різниці Δ , відстаней до скінченної кількості точок, з'єднаних у цикл:

$$\beta = \Delta_{i=1}^n d_i = \emptyset. \quad (2.6)$$

У формулі (2.6) кожна координата вектора представляє собою множину, побудовану на основі універсуму з p примітивних елементів:

$$\begin{aligned} a_j &= \{a_1, \dots, a_r, \dots, a_m\}, m = 2^p: \\ aDb &= \{0D0 = \square, 0D1 = x, 0Dx = 1, 0D\square = 0, 1D1 = \square, \\ 1Dx &= 0, 1D\square = 1, xDx = \square, xD\square = x, \square D\square = \square\}. \end{aligned} \quad (2.7)$$

Той факт, що симетрична різниця, обчислена покоординатно за допомогою теоретико-множинних операцій, дає порожній вектор, свідчить про те, що всі компоненти (відстані) в рівнянні відіграють однакову роль, і єдина операція, що застосовується до координат, є ключовою $d_{i,j}Dd_{i+1,j}$ визначена на сукупності $A = \{0, 1, x, \emptyset\}$, $x = \{0, 1\}$. Щоб визначити, чи є теоретико-множинний алфавіт замкнутим, можна використовувати Δ -операцію (яка є узагальненням XOR для множин). Алфавіт замкнутий, якщо застосування Δ -операції до всіх його символів призводить до порожньої множини: $0\Delta 1\Delta x\Delta \emptyset = \emptyset$. При двійковому кодуванні xor-сума буде дорівнювати нуль вектору: $01\oplus 10\oplus 11\oplus 00 = 00$.

Процедура обчислення векторного критерію якості залежить від значности алфавіту:

$$Q' = \begin{cases} m \oplus A \leftarrow \{m_i, A_i\} \in \{0, 1\}; \\ m \Delta A \leftarrow \{m_i, A_i\} \in \{0, 1, x\}. \end{cases} \quad (2.8)$$

У n -вимірному векторному логічному просторі критерій якості Q дозволяє чітко визначити три характеристики взаємодії між будь-якими двома об'єктами: відстань між ними та дві функції, що описують ступінь їхньої приналежності до певних множин. Алгоритм пошуку оптимального розв'язку серед кількох альтернатив, що характеризується мінімальною кількістю одиничних координат, проілюстровано на рис. 2.2.

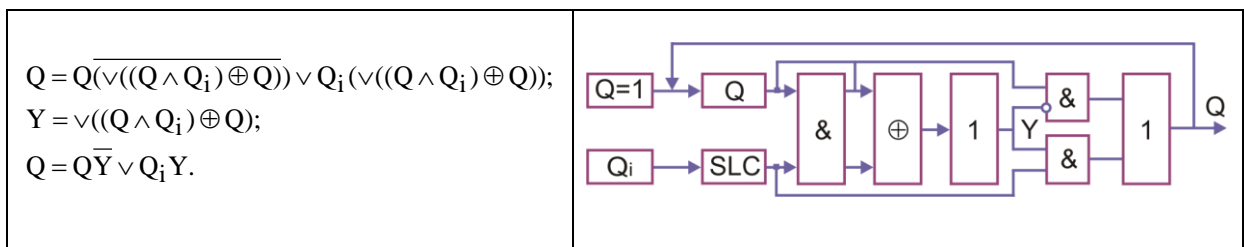


Рисунок 2.2 – Схема процес-моделі для вибору розв'язку

Вона включає наступні операції:

1. Ініціалізація та зсув: вектор-результат, призначений для зберігання оптимального розв'язку, спочатку заповнюється одиницями у кожній координаті, представляючи таким чином початковий, найменш ефективний розв'язок. Паралельно з цим, над поточним вектором Q виконується операція циклічного зсуву вліво з ущільненням одиниць (slc).

2. Порівняння векторів: Здійснюється порівняння вектора Q з черговою оцінкою розв'язку Q_i , отриманою зі списку доступних розв'язків.

3. Виконується векторна операція AND ($Q \wedge Q_i$) з наступним порівнянням результату з вектором Q , що дозволяє змінити його за умови, що вектор Q_i має менше число одиничних значень.

4. N -кратне повторення процедури пошуку оцінки кращого розв'язку.

Після застосування операції XOR, or -редукція перетворює n -розрядний вектор критерію якості на єдиний біт, використовуючи логічне "АБО". Це дозволяє швидко оцінити якість взаємодії об'єктів у кіберпросторі, оскільки логічні операції "АБО" можуть виконуватися паралельно. Така швидкість

особливо важлива для хмарних систем управління трафіком, що працюють у реальному часі.

2.3 Структура зв'язків компонентів системи

Різноманіття структур визначається їх топологією зв'язків або відношень між їх компонентами різної природи створює. У цьому контексті оптимізація кількості компонентів чи зв'язків набуває ключового значення для забезпечення ефективного функціонування системи, особливо з урахуванням її мети та існуючих обмежень. На цьому фоні висувається ідея про ефективність трикутних примітивів у формуванні транзакційних структур. Тріади компонентів відіграють домінуючу роль як у фізичній, так і в соціальній природі, виступаючи однією з найпоширеніших структур. Вони характеризуються наступними властивостями:

1. Властивість трикутного замикання, що забезпечує узгодженість вимірювань, лежить в основі будь-якої системи кількісної оцінки процесів і явищ.

2. Трикутник – це базова геометрична форма на площині, а тетраедр, утворений трикутними гранями, є його аналогом у тривимірному просторі.

3. Будь-яку фігуру на площині можна представити як комбінацію трикутників, а в тривимірному просторі – як комбінацію тетраедрів.

4. У трикутнику всі вершини суміжні, що означає наявність мінімального шляху між будь-якими двома вершинами — властивість, яка відсутня в інших геометричних примітивах.

5. Трикутник демонструє мінімальну вартість утворення міжз'єднань на площині, що дозволяє брати її за умовну одиницю.

6. Комунікації, засновані на трикутних структурах, є оптимальними за показником: відношення середньої довжини шляху між точками до вартості їхньої реалізації, що також забезпечує перевагу над іншими геометричними примітивами.

Таким чином, трикутник виступає унікальною структурною одиницею з точки зору ефективності побудови систем та комунікацій, що обумовлює його значущість у різних сферах застосування.

На підставі введеної ієрархії можна формувати квазі оптимальний кіберпростір. Він не є прив'язаним до геометричної топології поверхні і в цьому сенсі є віртуальним. Інформаційна структура CyberSpace має бути побудована у вигляді ієрархічної, замкненої системи як на глобальному, так і на локальному рівні, незалежно від конкретного етапу ієрархії. Основною одиницею структури цього простору пропонується зробити трикутну комірку. Такий підхід дозволяє суттєво скоротити обсяги інформації всередині CyberSpace, максимально досягнувши зменшення на третину. У результаті продуктивність усіх пристроїв прийому, передачі та зберігання контенту на планеті може зрости на 33%. Ця концепція ґрунтується на аксіомі транзитивного замикання кінцевого набору точок $(1, 2, 3, 4, \dots, n)$ в векторно-логічному просторі:

$$\begin{aligned}
 &1) d_1 = 0; \\
 &2) d_1 \oplus d_2 = 0; \\
 &3) d_1 \oplus d_2 \oplus d_3 = 0; \\
 &4) d_1 \oplus d_2 \oplus d_3 \oplus d_4 = 0; \\
 &5) d_1 \oplus d_2 \oplus \dots \oplus d_i \oplus \dots \oplus d_n = 0.
 \end{aligned}
 \tag{2.9}$$

Відновлення третьої сторони трикутника за двома відомими найбільш очевидне і ефективне. У цьому випадку достатньо передати через канал зв'язку лише дві сторони для визначення третьої, що забезпечує скорочення обсягу переданої інформації на 33%. В інших варіантах зменшення обсягів даних буде менш значним. Наприклад, у випадку замкнутого чотирикутника необхідно передати три відстані або сторони, щоб відновити четверту, що дає вигоду лише 25%. Можлива площинна інтерпретація просторової моделі CyberSpace представлена графічно на рис. 2.3.

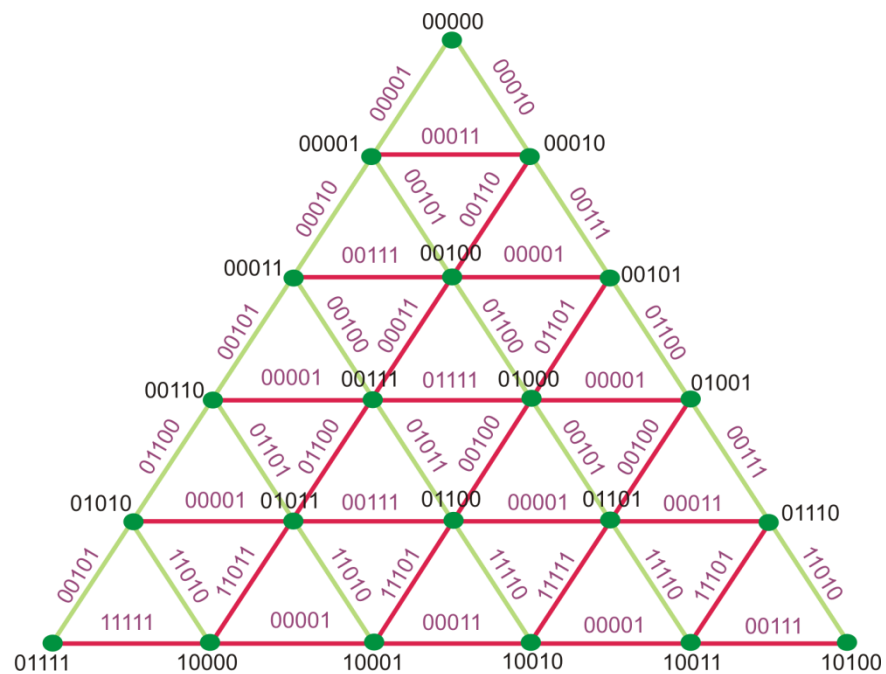


Рисунок 2.3—Схема трикутної топології кіберпростору

Для відновлення трикутної структури не обов'язково знати 2/3 обсягу інформації, що утворена кожною парою сторін трикутників. Досить мати сторони, що помічені зеленим кольором на рис. 2.3. Це становить 44% від обсягу відстаней кодів. Щодо стосуються інших 56% складових трикутників, то їх можна визначити аксіоматично, а саме:

$$d_1 \oplus d_2 \oplus d_3 = 0 \rightarrow d_3 = d_1 \oplus d_2. \quad (2.10)$$

Співвідношення сторін відновлювального трикутного простору є функцією від кількості шарів n трикутної структури:

$$\eta = \frac{\sum_{i=1, n} (i+1)}{\sum (3 \times i)} = \frac{1}{3} \left(\frac{2}{n+1} + 1 \right) = \frac{1}{3} \cdot \frac{2+n+1}{n+1} = \frac{1}{3} \cdot \frac{n+3}{n+1} = \frac{n+3}{3(n+1)}. \quad (2.11)$$

На основі графіка (рис. 2.4), для трикутного кіберпростору спостерігається тенденція до 33% граничного значення відношення обсягу переданої інформації до загальної кількості відстаней. Це означає, що для

достатньо великих трикутних просторів, відновлення повної інформації можливо шляхом передачі лише третини всіх відстаней (кодів).



Рисунок 2.4—Крива залежності кількості відстаней для відновлення простору

Трикутна метрика простору є найбільш ефективною з точки зору економічності. Її механізм транзитивного замикання забезпечує коротші відстані та маршрути між об'єктами. Завдяки сусідству кожного об'єкта з шістьма точками, трикутна топологія перевершує «манхеттенську» в аспекті оптимального прокладання шляхів між двома точками. Для мультипроцесорної системи оптимальна структура має бути побудована з трикутників. Ефективність структури визначається співвідношенням середньої довжини шляху між будь-якими точками до кількості сторін, що формують вартість топології. В трикутній топології довжина маршруту залишається не гіршою, ніж у «Манхеттені», хоча загальне число зв'язків у кожному чотирикутнику включає одну додаткову діагональ. Це дозволяє забезпечити прямий доступ до шести сусідів, що критично важливо для мультипроцесорної взаємодії. При порівнянні з «Манхеттеном» із діагональними зв'язками, трикутний простір має однакову довжину шляху,

але на один зв'язок менше. Однак «манхеттенська топологія» з другою діагоналлю чотирикутника розширює сусідство до восьми вершин. Оскільки трикутник є найбільш базовою геометричною фігурою для формування площини, його структура дозволяє досягнути високої ефективності в відновленні простору. У трикутнику вага однієї сторони, яку можна не враховувати для ідентифікації простору, є максимальною. Система трикутників надає можливість зберегти до 66% структурних компонентів, що не потребують опису при відтворенні простору, що значно полегшує формування Cyber Space. Інші примітивні площинні фігури демонструють менший вигравш в цьому аспекті. Здатність відновлювати до 66% несправних компонентів без втрати інформації є надзвичайно важливою для підвищення надійності цифрових систем. Це відкриває перспективи для створення стійких і ефективних архітектур просторових систем.

2.4 Якість топології комп'ютерної системи

Ефективність обміну даними між елементами мережі (будь то фізична мережа чи цифрова схема) значною мірою залежить від середньої відстані між цими елементами. При проектуванні топології обчислювальної системи важливо оцінити загальну віддаленість між усіма парами компонентів, розраховану як сума відстаней між ними, поділена на кількість з'єднань (ребер або дуг) у відповідному графі. Особливу увагу привертають прості геометричні форми, такі як чотирикутник, трикутник і тетраедр. Тетраедр відрізняється тим, що кожна його вершина з'єднана з трьома іншими, на відміну від трикутника, де кожна вершина має лише двох сусідів на площині.

Вартість інформаційної транзакції між компонентами системи, представлена графом з n вершин, є основою для критерію оцінки якості структури цього графу як знаходження $\min P_i$ між усіма парами вершин $\frac{1}{2}(n^2 - n)$ до кількості ребер топології k :

$$L = \frac{\sum_{i=1}^{\frac{1}{2}(n^2-n)} \min P_i}{\frac{1}{2}(n^2-n)} \times \frac{\frac{1}{2}(n^2-n)}{k} = \frac{\sum_{i=1}^{\frac{1}{2}(n^2-n)} \min P_i}{k} = \frac{1}{k} \sum_{i=1}^{\frac{1}{2}(n^2-n)} \min P_i, \quad (2.12)$$

де k є аргумент, що визначає кількість ребер в графі. Він є вартісною характеристикою топологічної структури (витратна частина) та розташований у знаменнику. Чим більша ця величина, тим менша сума найкоротших шляхів між усіма парами вершин графа, що наближається до мінімуму. Відповідно, низька інтегральна вартість транзакції L означає високу ефективність топологічної структури.

Приклад 2.1. Розрахувати критерії якості для трьох структур з 4 вершини (a, b, c, d): $\mathbf{G}_1 = \{(a,b), (b,c), (a,d), (c,d)\}$, $\mathbf{G}_2 = \{(a,b), (a,c), (a,d), (c,d), (b,c), (b,d)\}$, $\mathbf{G}_3 = \{(a,b), (b,c), (a,c), (a,d), (c,d)\}$:

$$\begin{array}{c}
 \mathbf{G}_1 = \mathbf{b} \\
 \mathbf{c} \\
 \mathbf{d}
 \end{array}
 \begin{array}{c|cccc}
 & \mathbf{a} & \mathbf{b} & \mathbf{c} & \mathbf{d} \\
 \hline
 \mathbf{a} & . & 1 & 1 & . \\
 \mathbf{b} & 1 & . & . & 1 \\
 \mathbf{c} & 1 & . & . & 1 \\
 \mathbf{d} & . & 1 & 1 & .
 \end{array}
 \mathbf{G}_2 = \mathbf{b}
 \begin{array}{c|cccc}
 & \mathbf{a} & \mathbf{b} & \mathbf{c} & \mathbf{d} \\
 \hline
 \mathbf{a} & . & 1 & 1 & 1 \\
 \mathbf{b} & 1 & . & 1 & 1 \\
 \mathbf{c} & 1 & 1 & . & 1 \\
 \mathbf{d} & 1 & 1 & 1 & .
 \end{array}
 \mathbf{G}_3 = \mathbf{b}
 \begin{array}{c|cccc}
 & \mathbf{a} & \mathbf{b} & \mathbf{c} & \mathbf{d} \\
 \hline
 \mathbf{a} & . & 1 & 1 & . \\
 \mathbf{b} & 1 & . & 1 & 1 \\
 \mathbf{c} & 1 & 1 & . & 1 \\
 \mathbf{d} & . & 1 & 1 & .
 \end{array}$$

За формулою (2.12) визначається відношення суми довжин мінімальних шляхів для кожної пари: $\{(a,b), (a,c), (a,d), (c,d), (b,c), (b,d)\}$, які з'єднані $\in \{4,6,5\}$ ребрами топології:

$$L_1 = \frac{1+2+1+1+2+1}{4} = \frac{8}{4} = 2,0;$$

$$L_2 = \frac{1+1+1+1+1+1}{6} = \frac{6}{6} = 1,0;$$

$$L_3 = \frac{1+1+1+1+2+1}{5} = \frac{7}{5} = 1,4.$$

L -критерій реагує на додавання ребер, тому навіть невеликі зміни в графі можуть суттєво вплинути на його оцінку. Оптимізація графових структур є критично важливою для ефективного планування та модернізації

транспортних мереж міст, враховуючи їхню привабливість для бізнесу та інвесторів.

Приклад 2.2. Проаналізувати критерію якості для оцінки графових структур з шістьма вершинами та відповідними топологіями їх з'єднання – трикутної, прямокутної, прямокутної з перехрестями (рис. 2.5).

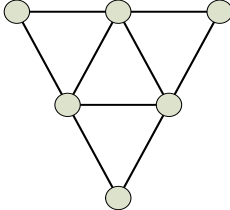
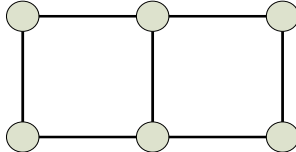
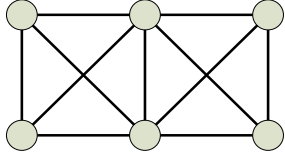
$G_1 =$	<table border="1"> <thead> <tr><th></th><th>a</th><th>b</th><th>c</th><th>d</th><th>e</th><th>f</th></tr> </thead> <tbody> <tr><th>a</th><td>.</td><td>1</td><td>1</td><td>.</td><td>.</td><td>.</td></tr> <tr><th>b</th><td>1</td><td>.</td><td>1</td><td>1</td><td>.</td><td>1</td></tr> <tr><th>c</th><td>.</td><td>1</td><td>.</td><td>.</td><td>.</td><td>1</td></tr> <tr><th>d</th><td>1</td><td>1</td><td>.</td><td>.</td><td>1</td><td>1</td></tr> <tr><th>e</th><td>.</td><td>.</td><td>.</td><td>1</td><td>.</td><td>1</td></tr> <tr><th>f</th><td>.</td><td>1</td><td>1</td><td>1</td><td>1</td><td>.</td></tr> </tbody> </table>		a	b	c	d	e	f	a	.	1	1	.	.	.	b	1	.	1	1	.	1	c	.	1	.	.	.	1	d	1	1	.	.	1	1	e	.	.	.	1	.	1	f	.	1	1	1	1	.	$G_2 =$	<table border="1"> <thead> <tr><th></th><th>a</th><th>b</th><th>c</th><th>d</th><th>e</th><th>f</th></tr> </thead> <tbody> <tr><th>a</th><td>.</td><td>1</td><td>.</td><td>1</td><td>.</td><td>.</td></tr> <tr><th>b</th><td>1</td><td>.</td><td>1</td><td>.</td><td>1</td><td>.</td></tr> <tr><th>c</th><td>.</td><td>1</td><td>.</td><td>.</td><td>.</td><td>1</td></tr> <tr><th>d</th><td>1</td><td>.</td><td>.</td><td>.</td><td>1</td><td>.</td></tr> <tr><th>e</th><td>.</td><td>1</td><td>.</td><td>1</td><td>.</td><td>1</td></tr> <tr><th>f</th><td>.</td><td>.</td><td>1</td><td>.</td><td>1</td><td>.</td></tr> </tbody> </table>		a	b	c	d	e	f	a	.	1	.	1	.	.	b	1	.	1	.	1	.	c	.	1	.	.	.	1	d	1	.	.	.	1	.	e	.	1	.	1	.	1	f	.	.	1	.	1	.	$G_3 =$	<table border="1"> <thead> <tr><th></th><th>a</th><th>b</th><th>c</th><th>d</th><th>e</th><th>f</th></tr> </thead> <tbody> <tr><th>a</th><td>.</td><td>1</td><td>.</td><td>1</td><td>1</td><td>.</td></tr> <tr><th>b</th><td>1</td><td>.</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><th>c</th><td>.</td><td>1</td><td>.</td><td>.</td><td>1</td><td>1</td></tr> <tr><th>d</th><td>1</td><td>1</td><td>.</td><td>.</td><td>1</td><td>.</td></tr> <tr><th>e</th><td>1</td><td>1</td><td>1</td><td>1</td><td>.</td><td>1</td></tr> <tr><th>f</th><td>.</td><td>1</td><td>1</td><td>.</td><td>1</td><td>.</td></tr> </tbody> </table>		a	b	c	d	e	f	a	.	1	.	1	1	.	b	1	.	1	1	1	1	c	.	1	.	.	1	1	d	1	1	.	.	1	.	e	1	1	1	1	.	1	f	.	1	1	.	1	.
	a	b	c	d	e	f																																																																																																																																																		
a	.	1	1	.	.	.																																																																																																																																																		
b	1	.	1	1	.	1																																																																																																																																																		
c	.	1	.	.	.	1																																																																																																																																																		
d	1	1	.	.	1	1																																																																																																																																																		
e	.	.	.	1	.	1																																																																																																																																																		
f	.	1	1	1	1	.																																																																																																																																																		
	a	b	c	d	e	f																																																																																																																																																		
a	.	1	.	1	.	.																																																																																																																																																		
b	1	.	1	.	1	.																																																																																																																																																		
c	.	1	.	.	.	1																																																																																																																																																		
d	1	.	.	.	1	.																																																																																																																																																		
e	.	1	.	1	.	1																																																																																																																																																		
f	.	.	1	.	1	.																																																																																																																																																		
	a	b	c	d	e	f																																																																																																																																																		
a	.	1	.	1	1	.																																																																																																																																																		
b	1	.	1	1	1	1																																																																																																																																																		
c	.	1	.	.	1	1																																																																																																																																																		
d	1	1	.	.	1	.																																																																																																																																																		
e	1	1	1	1	.	1																																																																																																																																																		
f	.	1	1	.	1	.																																																																																																																																																		
а		б		в																																																																																																																																																				

Рисунок 2.5 – Структури сполук графових вершин: а – трикутна, б – прямокутна; в – прямокутна з перехрестями

На рис. 2.5 три типи графа (а, б, в) містять 9, 7 і 11 ребер відповідно. Підрахунок L-критерію за формулою(2.12) дає такі результати:

$$L(G_1) = \frac{9 \times 1 + 6 \times 2}{9} = \frac{21}{9} = 2,33;$$

$$L(G_2) = \frac{7 \times 1 + 6 \times 2 + 2 \times 3}{7} = \frac{25}{7} = 3,57;$$

$$L(G_3) = \frac{11 \times 1 + 4 \times 2}{11} = \frac{19}{11} = 1,73.$$

2.5 In-memoгукomp'ютинг автономного транспорту

У роботах [25, 31, 32, 34] надається нова технологія in-memory комп'ютингу на основі використання векторно-логічних механізмів, що може бути використана для моніторингу та керування автономними транспортними засобами в інфраструктурі національних хмарних сервісів. Розробка векторно-логічного комп'ютингу спрямована на створення енергоефективних та швидких систем керування, зокрема для автономних транспортних засобів. Цей підхід, що дозволяє ефективно обробляти дані моніторингу навколишнього середовища, спрощує тестування та діагностику логічних компонентів вбудованих систем. Завдяки використанню національної 6G-інфраструктури, стає можливим автоматизувати доставку вантажів автономним транспортом, мінімізуючи час та енерговитрати. Векторно-логічні методи добре інтегруються з технологіями штучного інтелекту та квантовими обчисленнями, а in-memory комп'ютинг на основі надлишкових даних відкриває шлях до економічної обробки великих обсягів інформації безпосередньо в місці їх зберігання. Це особливо актуально для транспортної галузі, де кожний транспортний засіб щодня генерує терабайти даних, необхідних для автономного керування.

2.6 Висновки до розділу 2

Наведено основні моделі, метрики та критерії для аналізу ефективності транспортної інфраструктури.

Розглядається критерій якості топології, який можна реалізувати у хмарному сервісі для транспортної інфраструктури. Трикутна топологія виявляється ефективним рішенням для організації взаємодії компонентів у кіберфізичних системах, особливо при обробці транзакцій. Її перевага полягає в економічному використанні простору завдяки трикутній метриці, яка мінімізує відстані між точками через транзитивне замикання.

Дослідження показують [1, 3, 4, 14-16], що трикутна топологія дозволяє скоротити довжину маршрутів на 33-66% порівняно з метрикою Манхеттена. Крім того, така організація з'єднань підвищує надійність обчислювальних систем та мереж до 66%, забезпечуючи резервування зв'язків: кожен компонент має щонайменше двох сусідів. Використання тривимірної трикутної тетраедричної топології ще більше оптимізує транзакційні маршрути для кіберфізичних об'єктів, таких як авіація, глобальні обчислення та 3D-SoC. Важливо, що трикутна інфраструктура забезпечує високу стійкість до відмов, дозволяючи відновити до 66% пошкоджених з'єднань без втрати даних, зберігаючи систему в робочому стані навіть після значних пошкоджень. Одним із недоліків може бути незвичність трикутної системи координат для людини, яка звикла до декартової метрики, де точка на площині визначається двома координатами замість трьох.

3 СИСТЕМА МОНІТОРИНГУ ТА УПРАВЛІННЯ ТРАНСПОРТНИМИ МАРШРУТАМИ НА ОСНОВІ ГРАФОВИХ СТРУКТУР

3.1 Модифікація алгоритму Дейкстри для окремих топологій

Модифікація алгоритму Дейкстри, спрямоване на визначення найкоротших шляхів від обраної вершини графа (відправної точки) до інших вершин (можливих пунктів призначення), пов'язане зі зменшенням кількості операцій, таких як додавання і порівняння, а також зі збереженням отриманих даних для подальших етапів розрахунків. Цього досягають завдяки процедурі встановлення міток у алгоритмі Дейкстри, що дозволяє знизити загальну складність виконуваних обчислень до $O(n^2)$. Відомо, що для визначення найкоротших шляхів між усіма парами вершин використовується алгоритм Флойда-Уоршелла, складність якого становить $O(n^3)$.

Задача 3.1. Для графа G_1 з кількістю вершин 21, пов'язаних трикутною топологією слід знайти всі найкоротші шляхи їх довжини з вершини v_1 (на рис. 3.1) до кожної з інших вершин.

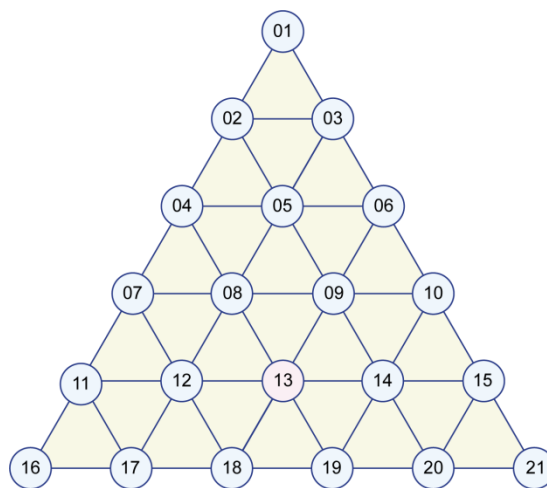


Рисунок 3.1 – Граф G_1 як топологія мережі

Розв'язок. Застосування алгоритму Дейкстри до неорієнтованого графа G_1 дозволяє знайти всі ланцюги найкоротшої довжини. У процесі

виконання алгоритму Дейкстри заповнюється табл. 3.1, де кількість рядків і стовпців дорівнює числу вершин графа $n=21$. При цьому заголовки рядків містять вершини, до яких потрібно знайти найкоротшу відстань.

Таблиця 3.1 – Реалізація алгоритму Дейкстри для графа G_1

G_1	$u=v_1$ $r=0$	$u=v_2$ $r=1$	$u=v_3$ $r=1$	$u=v_4$ $r=2$	$u=v_5$ $r=2$	$u=v_6$ $r=2$	$u=v_7$ $r=3$	$u=v_8$ $r=3$	$u=v_9$ $r=3$	$u=v_{10}$ $r=3$	$u=v_{11}$ $r=4$	$u=v_{12}$ $r=4$	$u=v_{13}$ $r=4$	$u=v_{14}$ $r=4$	$u=v_{15}$ $r=5$
v_2	$v_{1,1}$	-	-	-	-	-	-	-	-	-	-	-	-	-	-
v_3	$v_{1,1}$	$v_{1,1}$	-	-	-	-	-	-	-	-	-	-	-	-	-
v_4	$v_{1,\infty}$	$v_{2,2}$	$v_{2,2}$	-	-	-	-	-	-	-	-	-	-	-	-
v_5	$v_{1,\infty}$	$v_{2,2}$	$v_{2,2}$	$v_{2,2}$	-	-	-	-	-	-	-	-	-	-	-
v_6	$v_{1,\infty}$	$v_{1,\infty}$	$v_{3,2}$	$v_{3,2}$	$v_{3,2}$	-	-	-	-	-	-	-	-	-	-
v_7	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{4,3}$	$v_{4,3}$	$v_{4,3}$	-	-	-	-	-	-	-	-	-
v_8	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{4,3}$	$v_{4,3}$	$v_{4,3}$	$v_{4,3}$	-	-	-	-	-	-	-	-
v_9	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{5,3}$	$v_{5,3}$	$v_{5,3}$	$v_{5,3}$	-	-	-	-	-	-	-
v_{10}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{6,3}$	$v_{6,3}$	$v_{6,3}$	$v_{6,3}$	-	-	-	-	-	-
v_{11}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{7,4}$	$v_{7,4}$	$v_{7,4}$	$v_{7,4}$	-	-	-	-	-
v_{12}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{7,4}$	$v_{7,4}$	$v_{7,4}$	$v_{7,4}$	$v_{7,4}$	-	-	-	-
v_{13}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{8,4}$	$v_{8,4}$	$v_{8,4}$	$v_{8,4}$	$v_{8,4}$	-	-	-
v_{14}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{9,4}$	$v_{9,4}$	$v_{9,4}$	$v_{9,4}$	$v_{9,4}$	-	-
v_{15}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{10,4}$	$v_{10,4}$	$v_{10,4}$	$v_{10,4}$	$v_{10,4}$	-
v_{16}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{11,5}$	$v_{11,5}$	$v_{11,5}$	$v_{11,5}$	$v_{11,5}$
v_{17}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{11,5}$	$v_{11,5}$	$v_{11,5}$	$v_{11,5}$	$v_{11,5}$
v_{18}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{12,5}$	$v_{12,5}$	$v_{12,5}$	$v_{12,5}$
v_{19}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{13,5}$	$v_{13,5}$	$v_{13,5}$
v_{20}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{14,5}$	$v_{14,5}$
v_{21}	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{1,\infty}$	$v_{15,5}$

Алгоритм Дейкстри в цьому випадку підлягає такій модифікації: якщо на поточному етапі вже визначено кінцеву числову мітку, вона надалі не змінюється, оскільки може лише збільшитися. Отже, мінімальна відстань чи довжина шляху визначається цим числовим значенням мітки. Тому на наступних кроках немає сенсу виконувати процедури додавання та порівняння для тих вершин, для яких кінцеві мітки вже знайдено. Достатньо їх переглянути в порядку зростання індексів і обрати на кожній наступній ітерації як постійно помічені, доки всі вони не будуть вичерпані. Потім розглядаються вершини з кінцевими мітками зі значеннями на 1 більше. У табл. 3.1 міститься інформація щодо всіх найкоротших ланцюгів та їх

відповідних довжин, які зведено до табл. 3.2. Наприклад, найкоротший ланцюг з вершини v_1 у вершину v_{21} :

$$v_1 \xrightarrow{1} v_3 \xrightarrow{1} v_6 \xrightarrow{1} v_{10} \xrightarrow{1} v_{15} \xrightarrow{1} v_{21}, \text{dist} = 5.$$

Таблиця 3.2 – Найкоротші ланцюги та їх довжини для графа G_1

Ланцюги	Довжина
$v_1 \xrightarrow{1} v_2, r_{\min} = 1$	$r(v_1, v_2) = 1$
$v_1 \xrightarrow{1} v_3, r_{\min} = 1$	$r(v_1, v_3) = 1$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4, r_{\min} = 2$	$r(v_1, v_4) = 2$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_5, r_{\min} = 2$	$r(v_1, v_5) = 2$
$v_1 \xrightarrow{1} v_3 \xrightarrow{1} v_6, r_{\min} = 2$	$r(v_1, v_6) = 2$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_7, r_{\min} = 3$	$r(v_1, v_7) = 3$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_8, r_{\min} = 3$	$r(v_1, v_8) = 3$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_5 \xrightarrow{1} v_9, r_{\min} = 3$	$r(v_1, v_9) = 3$
$v_1 \xrightarrow{1} v_3 \xrightarrow{1} v_6 \xrightarrow{1} v_{10}, r_{\min} = 3$	$r(v_1, v_{10}) = 3$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_7 \xrightarrow{1} v_{11}, r_{\min} = 4$	$r(v_1, v_{11}) = 4$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_7 \xrightarrow{1} v_{12}, r_{\min} = 4$	$r(v_1, v_{12}) = 4$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_8 \xrightarrow{1} v_{13}, r_{\min} = 4$	$r(v_1, v_{13}) = 4$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_5 \xrightarrow{1} v_9 \xrightarrow{1} v_{14}, r_{\min} = 4$	$r(v_1, v_{14}) = 4$
$v_1 \xrightarrow{1} v_3 \xrightarrow{1} v_6 \xrightarrow{1} v_{10} \xrightarrow{1} v_{15}, r_{\min} = 4$	$r(v_1, v_{15}) = 4$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_7 \xrightarrow{1} v_{11} \xrightarrow{1} v_{16}, r_{\min} = 5$	$r(v_1, v_{16}) = 5$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_7 \xrightarrow{1} v_{11} \xrightarrow{1} v_{17}, r_{\min} = 5$	$r(v_1, v_{17}) = 5$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_7 \xrightarrow{1} v_{12} \xrightarrow{1} v_{18}, r_{\min} = 5$	$r(v_1, v_{18}) = 5$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_4 \xrightarrow{1} v_8 \xrightarrow{1} v_{13} \xrightarrow{1} v_{19}, r_{\min} = 5$	$r(v_1, v_{19}) = 5$
$v_1 \xrightarrow{1} v_2 \xrightarrow{1} v_5 \xrightarrow{1} v_9 \xrightarrow{1} v_{14} \xrightarrow{1} v_{20}, r_{\min} = 5$	$r(v_1, v_{20}) = 5$
$v_1 \xrightarrow{1} v_3 \xrightarrow{1} v_6 \xrightarrow{1} v_{10} \xrightarrow{1} v_{15} \xrightarrow{1} v_{21}, r_{\min} = 5$	$r(v_1, v_{21}) = 5$

На рис. 3.2 зображено дерево найкоротшихланцюгів з вершинив₁.

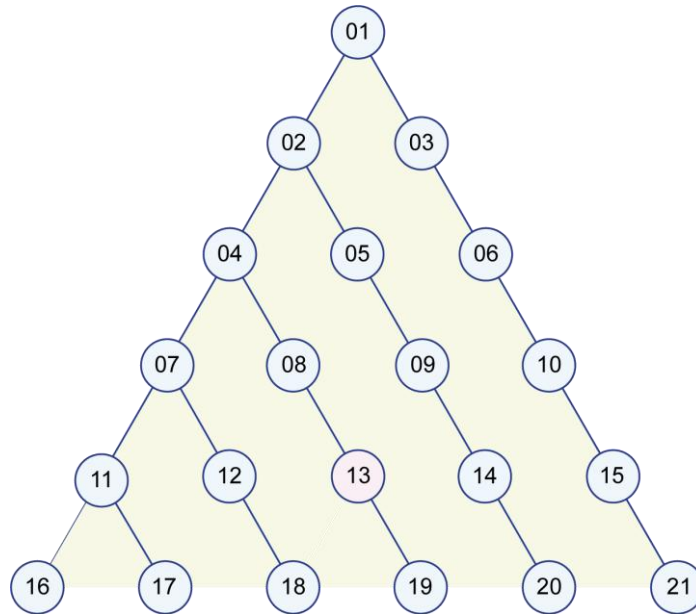
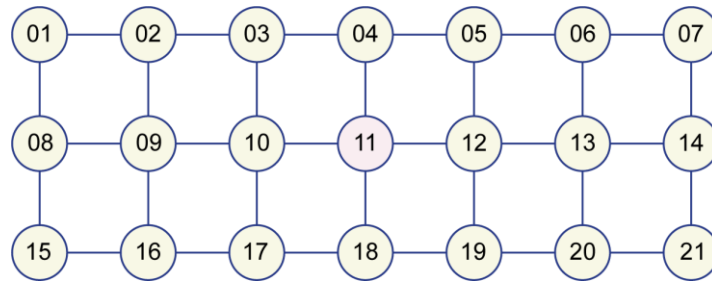


Рисунок 3.2 – Дерево найкоротшихланцюгів з вершинив₁

Оскільки ребра в графі G_1 мають вагові коефіцієнти 1 (див. рис. 3.2), з табл. 3.1 можна дійти висновку, що відстані при обчисленні можуть збільшуватися лише на одиницю. Таким чином, нескінченна мітка змінюється на кінцеву числову та потім вона більше не змінюється, тобто не може зменшитися. Це вказує на те, що відповідну найкоротшу відстань між вершинами вже визначено. У такому випадку кількість операцій додавання і порівняння в алгоритмі Дейкстри можна оптимізувати, виключивши перевірки з уже визначеними кінцевими числовими мітками у попередніх кроках. Крім того, варто послідовно обирати вершини зі списку з найменшими числовими мітками до їх вичерпання, а потім переходити до міток, які на одну одиницю більші. Таким підходом можна досягти скорочення часу виконання алгоритму приблизно на 30%.

Задача 3.2. Знайти найкоротші шляхи та їх довжини з вершинив₁ (рис. 3.3) до всіх вершин графа G_2 з прямокутною конфігурацією, що складається з 21 вершини.

Рисунок 3.3 – Топологія мережі G_2

Розв'язок. Поданий на рис. 3.3 граф G_2 є неорієнтованим. Алгоритм Дейкстри тут зазнає такої самої модифікації, що у задачі 3.1. Отже, з табл. 3.3 можна здобути інформацію щодо найкоротших ланцюгів та відповідних їм довжин.

Таблиця 3.3 – Найкоротші ланцюги з вершини v_1 графа G_2

Ланцюги	Довжина
$v_1 \rightarrow v_2, r_{\min} = 1$	$r(v_1, v_2) = 1$
$v_1 \rightarrow v_2 \rightarrow v_3, r_{\min} = 2$	$r(v_1, v_3) = 2$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4, r_{\min} = 3$	$r(v_1, v_4) = 3$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5, r_{\min} = 4$	$r(v_1, v_5) = 4$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6, r_{\min} = 5$	$r(v_1, v_6) = 5$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6 \rightarrow v_7, r_{\min} = 6$	$r(v_1, v_7) = 6$
$v_1 \rightarrow v_8, r_{\min} = 1$	$r(v_1, v_8) = 1$
$v_1 \rightarrow v_2 \rightarrow v_9, r_{\min} = 2$	$r(v_1, v_9) = 2$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_{10}, r_{\min} = 3$	$r(v_1, v_{10}) = 3$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_{11}, r_{\min} = 4$	$r(v_1, v_{11}) = 4$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_{12}, r_{\min} = 5$	$r(v_1, v_{12}) = 5$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6 \rightarrow v_{13}, r_{\min} = 6$	$r(v_1, v_{13}) = 6$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6 \rightarrow v_7 \rightarrow v_{14}, r_{\min} = 7$	$r(v_1, v_{14}) = 7$
$v_1 \rightarrow v_8 \rightarrow v_{15}, r_{\min} = 2$	$r(v_1, v_{15}) = 2$
$v_1 \rightarrow v_2 \rightarrow v_9 \rightarrow v_{16}, r_{\min} = 3$	$r(v_1, v_{16}) = 3$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_{10} \rightarrow v_{17}, r_{\min} = 4$	$r(v_1, v_{17}) = 4$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_{11} \rightarrow v_{18}, r_{\min} = 5$	$r(v_1, v_{18}) = 5$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_{12} \rightarrow v_{19}, r_{\min} = 6$	$r(v_1, v_{19}) = 6$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6 \rightarrow v_{13} \rightarrow v_{20}, r_{\min} = 7$	$r(v_1, v_{20}) = 7$
$v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6 \rightarrow v_7 \rightarrow v_{14} \rightarrow v_{21}, r_{\min} = 8$	$r(v_1, v_{21}) = 8$

Дерево найкоротших ланцюгів з вершини v_1 подано на рис. 3.4.

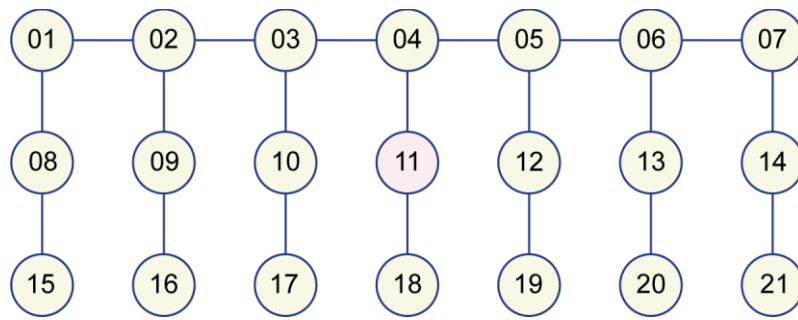


Рисунок 3.4 – Дерево найкоротших ланцюгів з вершини v_1 графа G_2

Таким чином, модифікація алгоритму Дейкстри має на меті підвищити ефективність обчислень у графах із ребрами одиничної ваги, а також її застосування в транспортній системі. Модифікація полягає в тому, що після ініціалізації та основного кроку алгоритму зміни стосуються лише тих вершин, які мають нескінченні мітки. Замість того щоб постійно перевіряти та оновлювати всі мітки, оновлюються виключно мітки для сусідніх вершин із нескінченними значеннями, які можуть отримати кінцеву числову мітку. Таким чином, знижується обсяг операцій порівняння та скорочується час обчислень. Водночас вводиться критерій вибору вершини для наступного оброблення: при відсутності ребер між постійно поміченою вершиною та вершинами з нескінченними мітками вибирається вершина з мінімальною кінцевою міткою серед поточного стовпця. Це дозволяє зосередитися на пошуку альтернативних маршрутів через інші вузли, не витрачаючи ресурси на зайві перевірки. Застосування алгоритму в транспортній інфраструктурі демонструє суттєві переваги. Як відзначено в дослідженнях, дорожні мережі, побудовані за принципом трикутних з'єднань між пунктами відправлення та прибуття, значно ефективніші за традиційні топології типу "манхеттен". Порівняльні показники продуктивності підтверджують [33], що такі трикутні мережі скорочують оцінку L з 22,3125 до 12,1.

Для впровадження цієї логіки у системи реального часу пропонується створення хмарної інфраструктури, що захищена від несанкціонованого

доступу. До її складу входять такі компоненти: хмари для моніторингу, управління та збереження історичних даних; блоки радіочастотної ідентифікації транспортних засобів; а також інтелектуальні сенсори дорожньої інфраструктури. Таким чином, забезпечується квазіоптимальне управління транспортом та дорожнім рухом у динамічному режимі.

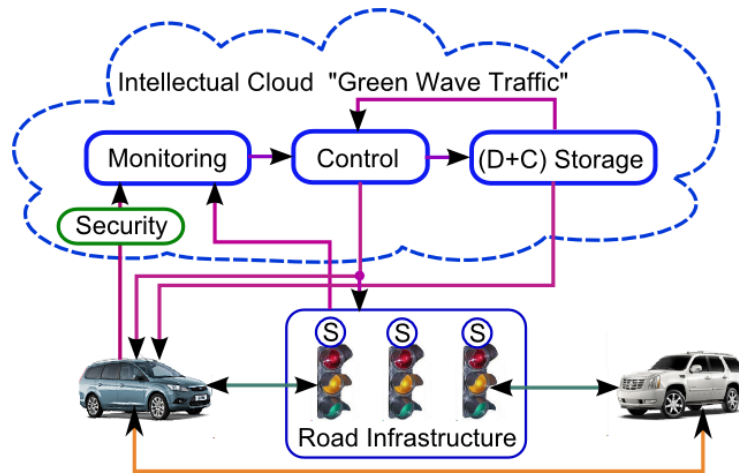


Рисунок 3.5 – Схематичне подання взаємозв'язків інтелектуальної хмари «Зелена хвиля» [33]

3.2 Сервіси хмарного керування транспортом

Структура масштабованої інтелектуальної кіберфізичної системи CloudTrafficControl (рис. 3.6) складається з таких основних елементів: 1) оцифровані правила та норми (законодавство); 2) цифрові ідентифікатори транспортних засобів; 3) кіберфізична інфраструктура для організації дорожнього руху, яка включає дороги, знаки, світлофори, парковки, заправні станції, ремонтні пункти, електронні карти, EmbeddedMicroSystems та SmartThings; 4) хмарна система управління дорожнім рухом; 5) "дорожня карта" сталого розвитку транспортної системи планети; 6) оцифровані транспортні процеси, що піддаються глобальній і локальній оптимізації за інтегральною метрикою, яка враховує витрати енергоресурсів, час доставки вантажів і пасажирів, а також фінансові й матеріальні затрати, пов'язані з проектуванням та експлуатацією кіберфізичної інфраструктури.

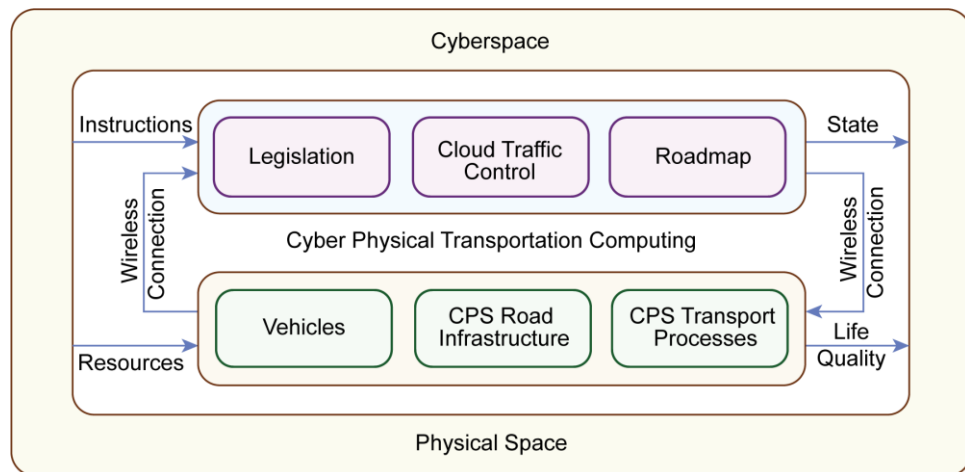


Рисунок 3.6 – Схема CyberPhysicalTransportationComputing

Хмарні кіберфізичні комп'ютингові сервіси включають (рис. 3.7): 1) онлайн хмарне управління дорожнім рухом з використанням цифрового моніторингу транспортних засобів і їх маршрутів; 2) хмарне управління світлофорами, як віртуальними, так і реальними, яке реагує на транспортний потік; 3) точний цифровий моніторинг та організаційне управління дорожнім рухом; 4) безперервний цифровий моніторинг експлуатації транспортного засобу; 5) забезпечення безперешкодного руху заданим маршрутом; 6) онлайн сервіс попереджень про потенційну небезпеку; 7) ергономічний сервіс, що надає відео- та аудіо-інформацію водієві онлайн; 8) персональний віртуальний кабінет водія; 9) захист персональних даних і аутентифікація користувача; 10) онлайн сервіс діагностики обладнання за запитом через віртуальний кабінет; 11) онлайн банківські послуги для товарів і послуг, які стосуються обслуговування автомобіля; 12) цифровий моніторинг і хмарне управління парковками з використанням вбудованих мікросистем та розумних пристроїв.

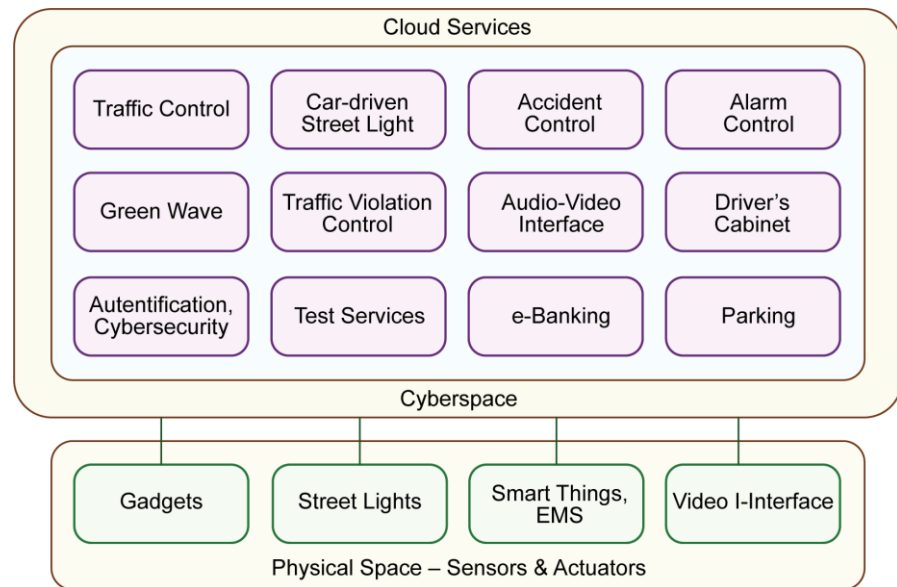


Рисунок 3.7 –Схема CloudTrafficControlServices

Архітектура для управління автономними транспортними засобами передбачає використання хмарних сервісів або локальних обчислювальних пристроїв з радіозв'язком(рис. 3.8). Важливо, що хмарні та бортові сервіси можуть взаємозамінно виконувати функції. Згідно з прогнозами Gartner[35] очікується зростання популярності національних хмарних інфраструктур, які забезпечують потреби внутрішнього ринку.

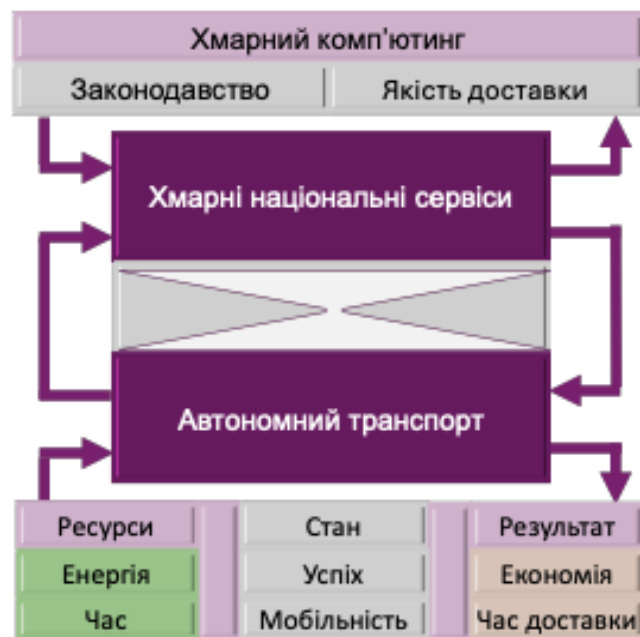


Рисунок 3.8 – Структура національного хмарного сервісу автономного транспорту

3.3 Висновки до розділу 3

Розроблено структуру системи моніторингу та управління транспортними маршрутами на основі графової топології для її подальшої інтеграції у хмарний сервіс, що складається з модуля планування оптимального маршруту на основі методу його побудови на дорожній інфраструктурі між двома координатами; покроковий процес створення алгоритму для вибору оптимального маршруту транспортного засобу включає використання модифікованого алгоритму Дейкстри, адаптованого до топологічної структури дорожньої інфраструктури. Це передбачає врахування специфічних умов та параметрів шляхів, таких як обмеження пропускної здатності, швидкісні параметри чи зони з підвищеним трафіком. Додатково проводиться оцінка різних типів графів на відповідність критеріям якості. Ця оцінка базується на аналізі матриць суміжності для розрахунку найкоротших відстаней між усіма парами вершин, а також визначенні загальної складності структури графа. Такий підхід дозволяє визначити слабкі та сильні сторони кожного типу графу. На основі порівняльного аналізу різних архітектур графів здійснюється підбір ефективних рішень для побудови оптимальних маршрутів. Це сприяє підвищенню продуктивності транспортної системи, забезпечуючи баланс між точністю розрахунків і швидкістю виконання алгоритму під час проектування нових дорожніх інфраструктур скорочення часу руху по замовленому маршруту.

ВИСНОВКИ

1. Проаналізовано технологічні тенденції автомобільної електроніки. Наведено аналіз сучасних досліджень та публікацій. Впровадження цифрових онлайн-сервісів хмарного інтелектуального трафіку тягне за собою позитивні технологічні та технічні трансформації та дає можливість автоматизувати процеси квазіоптимального управління транспортом в режимі реального часу.

2. Проаналізовано з точки зору ефективності основні моделі, метрики, критерії та електронну інфраструктуру, компоненти кіберфізичної системи моніторингу та управління дорожнім рухом у реальному режимі часу.

3. Визначено найкоротші шляхи та відстані з фіксованого компонента (вершини графа) до решти компонентів (вершин) інфраструктури. Для заданих матрицями суміжності різних типів графів з одиничними ваговими коефіцієнтами визначено найкоротші відстані між парами вершин та виконано оцінку інфраструктури за критеріями якості. Виконано порівняння різних архітектур та графових структур з метою визначення ефективних рішень.

4. Розроблено структуру системи моніторингу та управління транспортними маршрутами на основі графової топології для її подальшої інтеграції у хмарний сервіс, що складається з модуля планування оптимального маршруту на основі методу його побудови на дорожній інфраструктурі між двома координатами; покроковий синтез алгоритму побудови оптимального маршруту шляхом застосування модифікованого алгоритму Дейкстри на топологічній інфраструктурі дорожнього руху; оцінку різних конфігурацій графових топологій, що подані матрицями суміжності та діаграмою, за критеріями якості, шляхом знаходження найкоротших відстаней між усіма парами вершин і з урахуванням складності графа; порівняння різних графових архітектур для виявлення ефективних рішень для проектування нових дорожніх інфраструктур скорочення часу руху за заданими маршрутами. Система може бути імплементована до національних хмарних інфраструктур, які забезпечують потреби внутрішнього ринку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Hahanov V. Cloud-Driven Traffic Monitoring and Control Based on Smart Virtual Infrastructure [Text] / V. Hahanov, W.Gharibi, E. Litvinova, S. Chumachenko, A. Ziarmand, I. Englesi, I. Gritsuk, V. Volkov, A. Khakhanova // SAE Technical Paper. – USA. – 2017-01-0092, 2017. – 6 p. doi:10.4271/2017-01-0092.
2. The Pulse – Wireless BMS Design and Chipset Options. – May 28, 2021. – [https://octopart.com/pulse/p/wireless-bms-design-and-chipset-options]
3. Cloud Traffic Control System / [V.I.Hahanov, O.A.Guz, A.N.Ziarmand, Ngene Christopher Umerah, A.Arefjev] // 11th IEEE East-West Design and Test Symposium (EWDTS). – Rostov-on-Don. 27-30 September 2013: proceedings. – IEEE, 2013. – P.72-76. doi: 10.1109/EWDTS.2013.6673092.
4. Cloud traffic monitoring and control / [V.Hahanov, W.Gharibi, Baghdadi Ammar Awni Abbas, S.Chumachenko, O.Guz, E.Litvinova] // 2013 IEEE 7th International conference on intelligent data acquisition and advanced computing systems (IDAACS). – Berlin, Germany. – 12-14 September 2013: proceedings. – IEEE, 2013. – P. 244-248. – [10.1109/IDAACS.2013.6662681](https://doi.org/10.1109/IDAACS.2013.6662681)
5. Research on the Integrated Management of Highway Based on Radio Frequency Identification Technology / [Lu Antao, Li Yushan, Sun Yufang, Cao Chongzhen, Gao Kuigang, Xu Jing] // Third International Conference on Measuring Technology and Mechatronics Automation (ICMTMA). – Shanghai, China. – 6-7 January 2011: proceedings. – IEEE, 2011. – P. 116-119. doi: 10.1109/ICMTMA.2011.600.
6. Pandit A.A. RFID Tracking System for Vehicles (RTSV) / [A.A.Pandit, J.Talreja, A.K.Mundra] // First International Conference on Computational Intelligence, Communication Systems and Networks (CICSYN'09). – Indore, India. – 23-25 July 2009: proceedings. – IEEE, 2009. – P.160-165. doi: 10.1109/CICSYN.2009.50.

7. Improved Design of Vehicle Management System Based on RFID / [Jiang Lin-ying, Wang Shuai, Zhang Heng, Tan Han-qing]// International Conference on Intelligent System Design and Engineering Application (ISDEA).– Changsha, China. – 13-14 October 2010: proceedings.– IEEE, 2010.–P. 844-847. doi: 10.1109/ISDEA.2010.34.
8. Chen Xue-Mei. Vehicle management system based on multi-node RFID cards / Chen Xue-Mei, Wei Zhong-Hua// 30th Chinese Control Conference (CCC).– 22-24 July2011: proceedings.– IEEE, 2011. – P. 5497-5499.
9. Manikonda P.Intelligent traffic management system / P. Manikonda, A.K.Yerrapragada, S.S. Annasamudram// IEEE Conference on Sustainable Utilization and Development in Engineering and Technology (STUDENT).– Semenyih, Malaysia. – 20-21 October2011: proceedings.– IEEE,2011. –P. 119-122, doi: 10.1109/STUDENT.2011.6089337.
10. Wang, Q.Simulation study on the effect of automated driving in a road network environment / Q. Wang, L.Li, D. Hou, Z.Li, J. Hu// IET Intell. Transp. Syst., 2020. – 14: 228-232.<https://doi.org/10.1049/iet-its.2019.0395>
11. Zingirian N.Sensor clouds for Intelligent Truck Monitoring / N.Zingirian, C. Valenti// IEEE Intelligent Vehicles Symposium (IV).– Madrid, Spaine. –3-7 June2012: proceedings.– IEEE, 2012.– P. 999-1004.[10.1109/IVS.2012.6232192](https://doi.org/10.1109/IVS.2012.6232192)
12. An Intelligent Autonomous Vehicle Management System / [Branisso L.B., Kato E.R.R., Pedrino E.C., Morandin O., Tsunaki R.H.] // Second Brazilian Conference on Critical Embedded Systems (CBSEC). –SaoPaulo, Brazil. – 20-25 May2012: proceedings.– IEEE, 2012.– P. 42-47.doi: 10.1109/CBSEC.2012.23.
13. Brizgalov V.V. Architecture of traffic control systems using cloud computing / V.V.Brizgalov, V.Chukhantsev, E. Fedorkin// International Conference and Seminar on Micro/Nanotechnologies and Electron Devices (EDM).– 30 June– 4 July2010: proceedings.– IEEE, 2010.– P. 215-216.[10.1109/EDM.2010.5568826](https://doi.org/10.1109/EDM.2010.5568826)
14. Cyber physical system – smart cloud traffic control / [Vladimir Hahanov; Wajeb Gharibi; L. S. Abramova; Svetlana Chumachenko; Eugenia Litvinova; Anna Hahanova; Vladimir Rustinov; Vladimir Miz; Aleksey Zhalilo; Artur Ziarmand.] //

12th IEEE East-West Design & Test Symposium (EWDTS). – Kyiv, Ukraine. – 26-29 September 2014: proceedings. – IEEE, 2014. – P. 1–18. doi: 10.1109/EWDTS.2014.7027107.

15. Hahanov V. Intelligent road control and monitoring / [V. Hahanov, W. Gharibi, K. L. Man, E. Litvinova, S. Chumachenko, O. Guz] // In book: Jung, H.K., Kim, J., Sahama, T., Yang, C.H. (eds) Future Information Communication Technology and Applications. – Series: Lecture Notes in Electrical Engineering (LNEE). – Springer, Dordrecht. – 2013, Vol. 235. – P. 327-335. https://doi.org/10.1007/978-94-007-6516-0_36.

16. Gorobets A. Dijkstra Algorithm for cyber structures analysis / [A. Gorobets, I. Chugurov, D. Scherbin, S. Chumachenko] // 12th International Conference on the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM). – Lviv-Polyana, Ukraine. – 19-23 Feb. 2013: proceedings. – IEEE, 2013. – P. 61-65.

17. Dijkstra E. W. A note on two problems in connexion with graphs / Dijkstra E. W. // Numerische Mathematik. – 1959. – Vol. 1. – P. 269-271. <https://doi.org/10.1007/BF01386390>.

18. Моделі аналізу ефективності обчислювальних структур / [В.І. Хаханов, С.В. Чумаченко, Murad Ali Abbas, А.А. Горобец, М.В. Скоробогатий, В.В. Білоус] // Радіоелектроніка та інформатика. – 2012. – №3. – С. 66-71. [<https://openarchive.nure.ua/server/api/core/bitstreams/eedc1b40-b759-4898-805a-6b2dc6fd2311/content>]

19. The e-IRGSP4 project for e-IRG, Best Practices for the use of e-Infrastructures by large-scale research infrastructures. – 24 February 2015. – [<http://e-irg.eu/>].

20. Holsinger Sy. EGI: Implementing service management in a large-scale e-Infrastructure / Sy Holsinger, Sergio Andreozzi // 2014 IEEE Network Operations and Management Symposium (NOMS). – Krakow, Poland. – 05-09 May 2014: proceedings. – IEEE, 2014. – P. 1-5. doi: 10.1109/NOMS.2014.6838371.

21. Federating Infrastructure as a Service Cloud Computing System to Create a

Uniform E-infrastructure for Research / [D. C. H. Wallom, M. Turilli, M. Drescher, D. Scardaci, S. Newhouse] // *2015 IEEE 11th International Conference on e-Science*. – Munich, Germany. – 31 August – 4 September 2015: proceedings. – IEEE, 2015. – P. 155-164. doi: 10.1109/eScience.2015.51.

22. E-infrastructure: The ecosystem for innovation, 9 October 2013. [www.bis.gov.uk].

23. Towards an e-Infrastructure for Urban Research across Australia/ Richard O. Sinnott, Gerson Galang, Martin Tomko, Robert Stimson] // *IEEE 7th International Conference on E-Science (e-Science)*. – 2011. – <http://ieeexplore.ieee.org/document/6123291/>

Towards an e-Infrastructure for Urban Research across Australia / [R. O. Sinnott, G. Galang, M. Tomko and R. Stimson] // *2011 IEEE Seventh International Conference on e-Science*. – Stockholm, Sweden. – 5-8 December 2011: proceedings. – IEEE, 2012. – P. 295-302. doi: 10.1109/eScience.2011.48.

24. Drescher M. MS518 EGI Platform Roadmap // June 2014 [online] Available: <https://documents.egi.eu/document/2232>.

25. Cybersocial FML – computing II. Relations & Metrics / [V. Hahanov, S. Chumachenko, E. Litvinova et al.] // *East-West Design & Test Symposium (EWDTS'21)*. – Batumi, Georgia. – 10-13 September 2021: proceedings. – IEEE, 2021. – P. 207–212. <https://doi.org/10.1109/ewdts52692.2021.9581007>.

26. New England E-infrastructure: EPSRC e-Infrastructure strategy 2018. [<http://www.rcuk.ac.uk/research/xrcprogrammes/OtherProgs/eInfrastructure/>].

27. Large scale cloudlets deployment for efficient mobile cloud computing / [L. Tawalbeh, Y. Jararweh, F. Ababneh, F. Dosari] // *Journal of Networks*. – 2015. – Vol. 10, no. 01. – P. 70–76. doi: 10.4304/jnw.10.1.70-76.

28. Chronopoulos A.T. A real-time traffic simulation using a communication latency hiding parallelization/ A.T. Chronopoulos, C.M. Johnston // *IEEE Transactions on Vehicular Technology*. – 2002. – Vol. 51, Issue 3. – P. 498 – 510. DOI: 10.1109/TVT.2002.1002499.

29. Vertical Handover Scheme for Car-to-Car Communication Based on IEEE

802.21 Standard / [A. Widhiasi, V. Mohanan, M. F. Pasha and R. Budiarto] // *2010 Second International Conference on Computer Engineering and Applications*. – Bali, Indonesia. – 19-21 March 2010: proceedings. – IEEE, 2010. – P. 143-147. doi: 10.1109/ICCEA.2010.36.

30. A Study on Behavior of Autonomous Vehicles Cooperating with Manually-Driven Vehicles / [Y. Nishimura et al.] // *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. – Kyoto, Japan. – 11-15 March 2019: proceedings. – IEEE, 2019. – P. 212-219. doi: 10.1109/PERCOM.2019.8767390.

31. Structure and Metrics of Emerging Computing / [V. Hahanov, O. Shevchenko, V. H. Abdullayev et al.] // *15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*. – Lviv-Slavske, Ukraine. – 23-27 February 2020: proceedings. – IEEE, 2020. – P. 920–925. <https://doi.org/10.1109/tcset49122.2020.235571>.

32. Vector Models for Modeling Logic Based on XOR-Relations / [V. Hahanov, E. Litvinova, O. Shevchenko et al.] // *16th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*. – Lviv-Slavske, Ukraine. – 24-27 February 2022: proceedings. – IEEE, 2022. – P. 823–828. doi: 10.1109/tcset55632.2022.9766894.

33. Зіарманд А. Н. Моделі і методи кіберфізичного комп'ютерного моніторингу і хмарного управління транспортом: автореф. дис... канд. техн. наук : 05.13.05 – “Комп'ютерні системи і компоненти” / А. Н. Зіарманд. – Харків: ХНУРЕ, 2017. – 23 с.

34. Vector synthesis of fault testing map for logic / [V. Hahanov, W. Gharibi, S. Chumachenko, E. Litvinova] // *IAES International Journal of Robotics and Automation (IJRA)*. – September 2024. – Vol. 13, No. 3. – P. 293-306. doi: 10.11591/ijra.v13i3, pp293-306.

35. Chandrasekaran A. Spotlight on 2024 Gartner Hype Cycle™ for Emerging Technologies. October 10, 2024. [<https://www.gartner.com/en/articles/hype-cycle-for-emerging-technologies>]