

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління  
(повна назва)

Кафедра електронних обчислювальних машин  
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

Рівень вищої освіти другий (магістерський)

Метод прогнозування еластичного трафіка  
в мережах промислового Інтернету речей

(тема)

Виконав:

студент II курсу, групи СПМ-23-2  
Сирадоев А.О.  
(прізвище, ініціали)

Спеціальність 123 «Комп'ютерна інженерія»  
(код і повна назва спеціальності)

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

Освітня програма Системне програмування  
(повна назва освітньої програми)

Керівник: проф. Можєв О.О.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ комп'ютерної інженерії та управління \_\_\_\_\_

Кафедра \_\_\_\_\_ електронних обчислювальних машин \_\_\_\_\_

Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_

Спеціальність \_\_\_\_\_ 123 «Комп'ютерна інженерія» \_\_\_\_\_  
(код і повна назва)

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

Освітня програма \_\_\_\_\_ Системне програмування \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту \_\_\_\_\_ Сирадоєву Андрію Олександровичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи \_\_\_\_\_ Метод прогнозування еластичного трафіка в мережах  
промислового Інтернету речей \_\_\_\_\_

затверджена наказом по університету від “ 22 ” листопада 2024 р. № 1236 Ст

2. Термін подання студентом роботи до екзаменаційної комісії \_\_\_\_\_ 20 січня 2025 р.

3. Вхідні дані до роботи \_\_\_\_\_ операційна система –Windows або Linux,  
1Гб оперативної пам'яті ПК процесор на 1Ггц, 100Мб свобідної пам'яті на жорсткому  
диску \_\_\_\_\_

4. Перелік питань, що потрібно опрацювати у роботі \_\_\_\_\_

1) аналіз можливостей та завдань сучасних систем Інтернету речей \_\_\_\_\_

2) принципи організації та властивості еластичних систем \_\_\_\_\_

3) Короткострокове квантове прогнозування еластичного трафіку  
в мережах промислового Інтернету речей \_\_\_\_\_

4) висновки \_\_\_\_\_

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 15

---

---

---

---

---

---

---

---

---

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1 )

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

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної області	26.11.24-30.11.24	
2	Вибір та обґрунтування методики		
3	Вибір інструментальних засобів	02.12.24-05.12.24	
4	Розробка методу		
5	Проведення експериментів	11.12.24-21.12.24	
6	Оформлення матеріалів кваліфікаційної роботи	23.12.24-03.01.25 04.01.25-07.01.25	
7	Подання кваліфікаційної роботи її попередній захист	08.01.25-11.01.25	
8	Подання кваліфікаційної роботи на рецензування	13.01.25-17.01.25	

Дата видачі завдання 25 листопада 2024 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_  
(підпис)

проф. Можасєв О.О.  
(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 73 с., 22 рис., 2 дод., 20 джерел.

ІНТЕРНЕТ РЕЧЕЙ, ПРОМИСЛОВИЙ ІНТЕРНЕТ РЕЧЕЙ,  
ЕЛАСТИЧНИЙ ТРАФІК, ХМАРНЕ СЕРЕДОВИЩЕ.

Метою кваліфікаційної роботи є підвищення ефективності функціонування промислового Інтернету речей за рахунок розроблення підходу до завчасного горизонтального масштабування на основі прогнозування еластичного трафіка. Предметом дослідження є методи прогнозування еластичного трафіка в мережах промислового Інтернету речей. Об'єктом дослідження є процес горизонтального масштабування базової хмарної системи промислового Інтернету речей. Проведений аналіз можливостей та завдань сучасних систем Інтернету речей. Визначені принципи організації та властивості еластичних систем Розроблений підхід до короткострокового квантового прогнозування еластичного трафіку в мережах промислового Інтернету речей.

## ABSTRACT

Master's thesis: 73 pages, 22 figures, 2 appendices, 20 sources.

INTERNET OF THINGS, INDUSTRIAL INTERNET OF THINGS,  
ELASTIC TRAFFIC, CLOUD ENVIRONMENT.

The purpose of the qualification work is to increase the efficiency of the functioning of the Industrial Internet of Things by developing an approach to early horizontal scaling based on elastic traffic forecasting. The subject of the study is methods for forecasting elastic traffic in industrial Internet of Things networks. The object of the study is the process of horizontal scaling of the basic cloud system of the Industrial Internet of Things. An analysis of the capabilities and tasks of modern Internet of Things systems has been conducted. The principles of organization and properties of elastic systems have been determined. An approach to short-term quantum forecasting of elastic traffic in industrial Internet of Things networks has been developed.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	7
ВСТУП .....	8
1 АНАЛІЗ МОЖЛИВОСТЕЙ ТА ЗАВДАНЬ СУЧАСНИХ СИСТЕМ ІНТЕРНЕТУ РЕЧЕЙ .....	10
1.1 Роль Інтернету речей у сучасному світі.....	10
1.2 Класифікація Інтернету речей .....	14
1.3 Порівняння споживчого і промислового Інтернету речей .....	20
2 ПРИНЦИПИ ОРГАНІЗАЦІЇ ТА ВЛАСТИВОСТІ ЕЛАСТИЧНИХ СИСТЕМ .....	27
2.1 Концепція еластичної системи .....	27
2.2 Способи реалізації еластичних систем .....	31
2.3 Управління обчислювальними ресурсами еластичних систем .....	36
3 КОРОТКОСТРОКОВЕ КВАНТОВЕ ПРОГНОЗУВАННЯ ЕЛАСТИЧНОГО ТРАФІКУ В МЕРЕЖАХ ПРОМИСЛОВОГО ІНТЕРНЕТУ РЕЧЕЙ .....	44
3.1 Методи аналізу трафіку промислового Інтернету речей .....	44
3.2 Квантовий підхід до прогнозу еластичного трафіку .....	48
3.3 Приклад прогнозування ділянки агрегованого трафіка ПоТ .....	53
ВИСНОВКИ .....	56
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	58
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	60
ДОДАТОК Б Приклад застосування квантового прогнозування еластичного трафіку.....	69

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ  
І ТЕРМІНІВ

ВМ – віртуальна машина

ДТСЗ – довготривала статистична залежність

ІР – Інтернет речей

Мб – мегабайт

ОВ – обчислювальний вузол

ОЗП – оперативний запам'ятовуючий пристрій

ОС – операційна система

ПІР – промисловий Інтернет речей

ПК – персональний комп'ютер

ЦОД – центр обробки даних

ЕС – Edge Computing

IIoT – Industrial Internet of Things

IoT – Internet of Things

## ВСТУП

У наш час технології Інтернету речей поступово охоплюють все більше різноманітних галузей. Зокрема, окремим типом даної технології став промисловий Інтернет речей (ПоТ), який суттєво відрізняється від вже звичного побутового Інтернету речей (ІоТ). Більшість завдань щодо аналізу, обробки та зберігання даних ПоТ виконується у хмарному середовищі. Специфіка ПоТ визначила найбільш сприятливу характеристику базової хмарної системи для виконання завдань ПоТ – еластичність системи, яка дозволяє швидко реалізовувати горизонтальне масштабування ресурсів практично без зупинки та впливу обчислювального процесу, що є важливим для оперативних складових ПоТ. Однак при цьому можуть з'являтися черги на обробку запитів внаслідок деяких часових втрат на підключення додаткового реального або віртуального ресурсного обладнання.

Отже, спираючись на вищевикладене, науково-технічне завдання, спрямоване на розробку підходу до прогнозування еластичного трафіка в мережах промислового Інтернету речей є актуальним та доцільним.

Предметом дослідження є методи прогнозування еластичного трафіка в мережах промислового Інтернету речей.

Об'єктом дослідження є процес горизонтального масштабування базової хмарної системи промислового Інтернету речей.

Метою дослідження є підвищення ефективності функціонування промислового Інтернету речей за рахунок розроблення підходу до завчасного горизонтального масштабування на основі прогнозування еластичного трафіка.

Для досягнення мети повинні бути вирішені такі часткові завдання:

- 1 провести аналіз можливостей та завдань сучасних систем Інтернету речей;
- 2 визначити принципи організації та властивості еластичних систем;

З розробити підхід до короткострокового квантового прогнозування еластичного трафіку в мережах промислового Інтернету речей.

Практична цінність дослідження полягає в підвищенні продуктивності промислового Інтернету речей за рахунок розроблення підходу до завчасного горизонтального масштабування на основі прогнозування еластичного трафіка.

# 1 АНАЛІЗ МОЖЛИВОСТЕЙ ТА ЗАВДАНЬ СУЧАСНИХ СИСТЕМ ІНТЕРНЕТУ РЕЧЕЙ

## 1.1 Роль Інтернету речей у сучасному світі

У сучасному світі технології Інтернету речей (Internet of Things, IoT) набирають все більших обертів. Так, у 2023 році світовий ринок Інтернету речей активно зростає, кількість підключених пристроїв IoT досягла 16,7 млрд., що в 2 рази перевищує населення Землі.

А прогнози експертів ринку вказують на те, що до 2030 року кількість пристроїв, що використовуються, швидше за все, збільшиться до більш ніж 29 мільярдів пристроїв IoT (рисунок 1.1) [1].

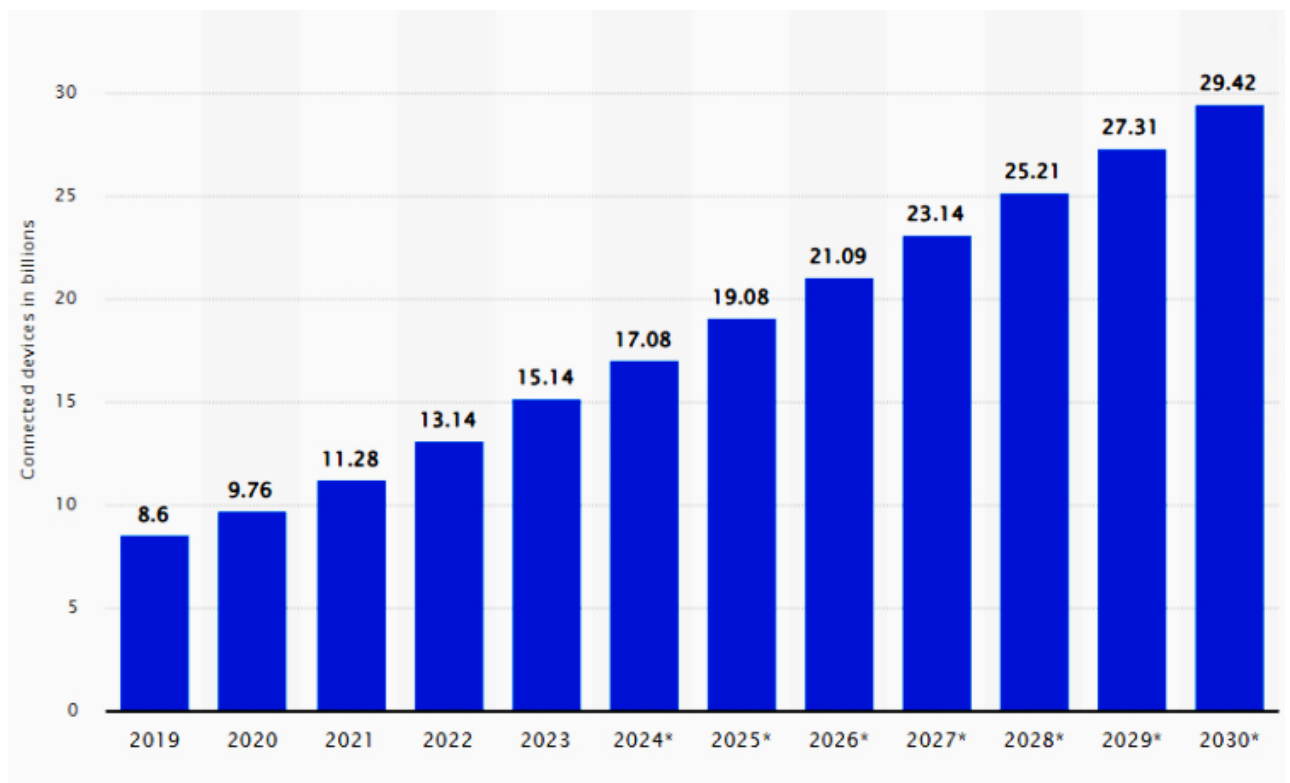


Рисунок 1.1 – Зростання кількості IoT-пристроїв у світі у 2019–2030 роках

Провідні компанії світу постійно збільшують витрати на розвиток IoT. Так, за прогнозами International Data Corporation (IDC), у поточному 2024

році європейські компанії готові витратити на IoT-продукти не менше 260 млрд доларів (рисунок 1.2) [2]. А це означає, що йдеться вже про стратегічний пріоритет для організацій, які стикаються із сучасними викликами.

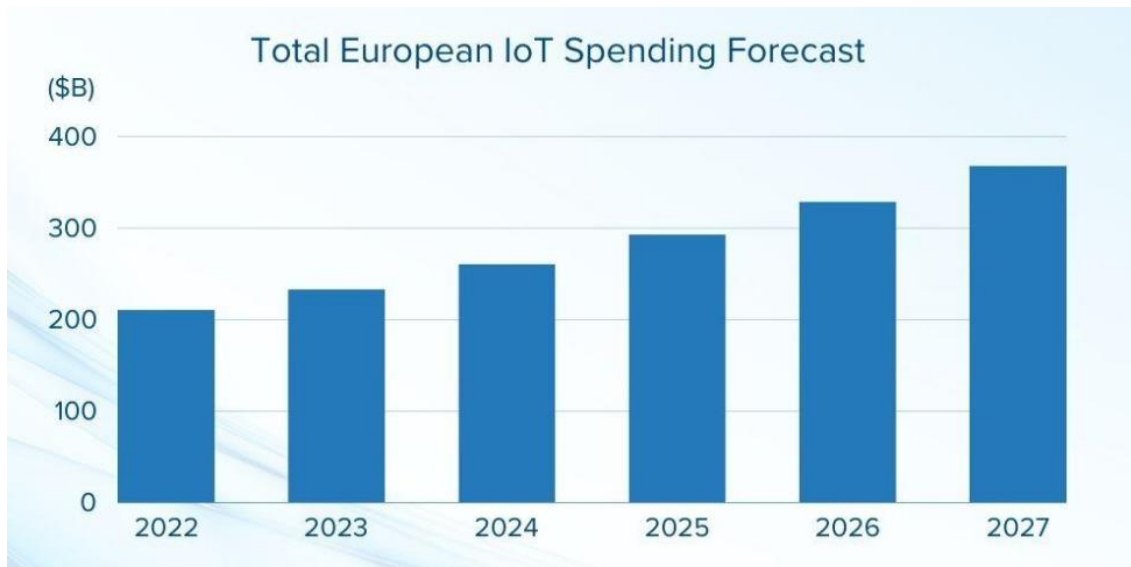


Рисунок 1.2 – Прогнозний обсяг сукупних витрат на IoT у Європі

Такі інвестиції з боку європейських компаній є лише частиною глобальної тенденції. Зважаючи на все, зовсім скоро йтиметься про цифри іншого порядку: до 2027 року глобальний обсяг продажів IoT для бізнесу може досягти 1,2 трлн доларів, а до 2030 року глобальний попит на IoT - рішення може збільшитися у 2,3 рази за шість років.

Інтернет речей можна розглядати як методологію обчислювальної мережі фізичних предметів («речей»), оснащених вбудованими технологіями для взаємодії один з одним або із зовнішнім середовищем, яка розглядає організацію таких мереж як явище, здатне перебудувати економічні та суспільні процеси, що виключає з частини дій та операцій необхідність безпосередньої участі людини.

Ідея Інтернету речей полягає в тому, щоб забезпечити різні предмети та пристрої датчиками, які збирають інформацію та передають її через мережу. Ці дані потім аналізуються та використовуються для прийняття рішень або

автоматизації процесів. Ключовий компонент IoT – це хмарні сервіси, які обробляють та зберігають великі обсяги даних, а також дозволяють керувати та контролювати пристрої віддалено (рисунок 1.3) [3].

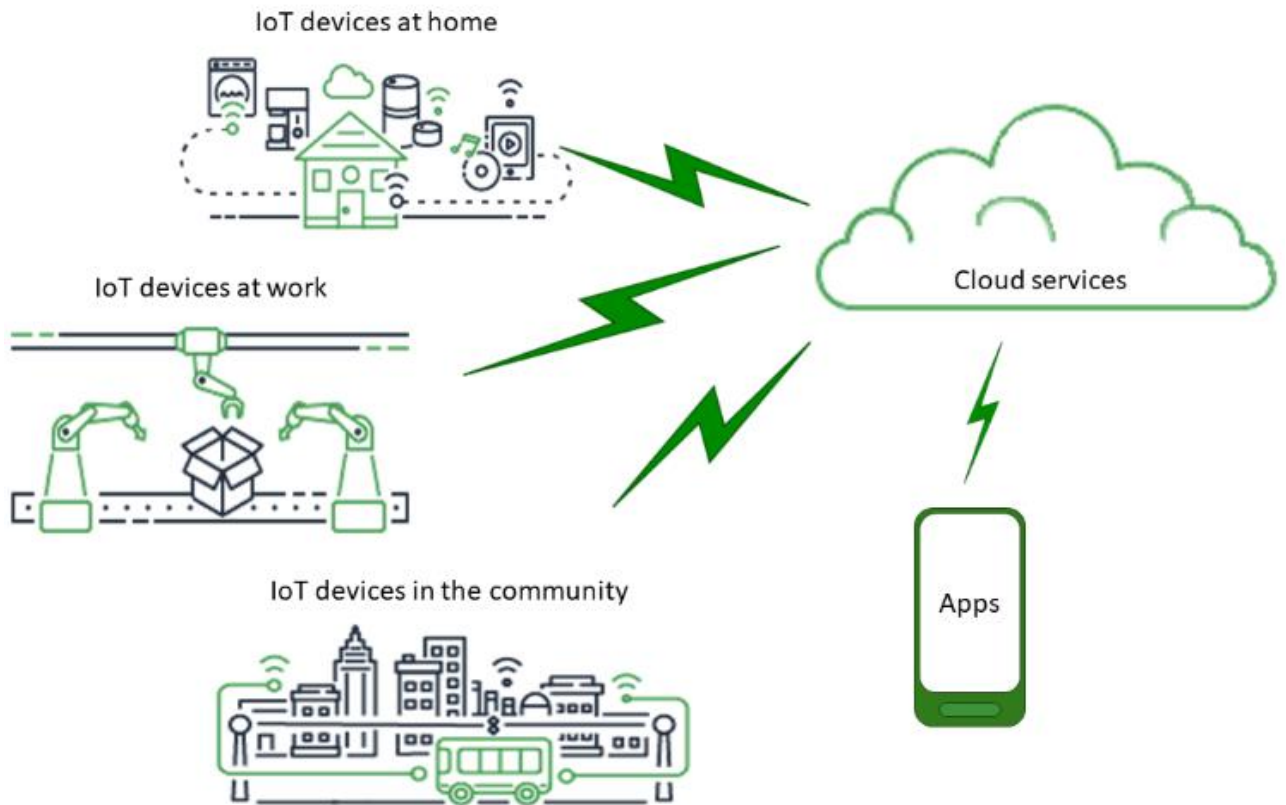


Рисунок 1.3 – Хмарний сервіс Інтернету речей

Концепція IoT сформульована у 1999 році як осмислення перспектив широкого застосування засобів радіочастотної ідентифікації для взаємодії фізичних предметів між собою та із зовнішнім оточенням. Наповнення концепції Інтернету речей різноманітним технологічним змістом та впровадження практичних рішень для її реалізації, почалося з 2010-х років та вважається стійкою тенденцією в інформаційних технологіях, насамперед завдяки повсюдному поширенню бездротових мереж, появі хмарних обчислень, розвитку технологій міжмашинної взаємодії, початку активного переходу на IPv6 і програмно-конфігуровані мережі [4]. Серед технологій та передумов, які найбільше сприяли розвитку Інтернету речей, варто виділити такі [5]:

- – загальний доступ до недорогих та надійних датчиків з низьким споживанням електроенергії;
- мініатюризація апаратного забезпечення;
- автоматична ідентифікація даних;
- широкий спектр мережних протоколів, що полегшують підключення;
- хмарні обчислювальні платформи;
- машинне навчання.

Задіяння в Інтернеті речей предметів фізичного світу, необов'язково оснащених засобами підключення до мереж передачі, вимагає застосування технологій ідентифікації цих предметів (речей). Хоча поштовхом для появи концепції стала технологія RFID (Radio Frequency Identification, рисунок 1.4) [6], але як такі технології можуть використовуватися всі засоби, що застосовуються для автоматичної ідентифікації: ідентифікатори, що оптично розпізнаються (штрих-коди, Data Matrix, QR-коди), засоби визначення місцезнаходження в режимі реального часу.



Рисунок 1.4 – Принцип роботи технології RFID, яка використовує радіочастотні сигнали, щоб розпізнавати та відстежувати об'єкти

При всеосяжному поширенні Інтернету речей важливо забезпечити унікальність ідентифікаторів об'єктів, що, своєю чергою, вимагає стандартизації. Для об'єктів, безпосередньо підключених до Інтернет-мереж, традиційний ідентифікатор – MAC-адреса мережного адаптера, що дозволяє ідентифікувати пристрій на каналному рівні, при цьому діапазон доступних адрес є практично обмеженим, а використання ідентифікатора каналного рівня не занадто зручно для програм. Більш широкі можливості для ідентифікації для таких пристроїв дає протокол IPv6, який збільшує розмір адреси до 128 бітів, що забезпечує практично необмежену кількість унікальних адрес мережного рівня (не менше 300 млн пристроїв IoT на одного жителя Землі).

Розглянута концепція Інтернету речей включає три важливих для функціонування компонента [7]:

- смарт-пристрій – електронний пристрій, як правило, пов'язаний з іншими пристроями або мережами за допомогою різних бездротових протоколів, таких як Bluetooth, NFC, Wi-Fi та ін.; смарт-пристрої можуть працювати в інтерактивному режимі та автономно;

- застосунок IoT – набір сервісів та іншого програмного забезпечення, які необхідні для об'єднання даних, які отримуються від IoT-пристроїв, зазвичай подібні програми використовують у роботі машинне навчання, і навіть штучний інтелект для аналізу отриманих матеріалів та подальшої інтелектуальної реакції;

- користувальницький графічний інтерфейс – інтерфейс, який допускає керування IoT-пристроями з боку людини.

## 1.2 Класифікація Інтернету речей

Технології IoT передбачають певну класифікацію. Якщо класифікувати системи Інтернету речей за загальним призначенням, то умовно розглянуту

концепцію можна розділити на дві великі категорії – споживчий Інтернет речей і промисловий (індустріальний) Інтернет речей (рисунок 1.5) [8].

Головна відмінність наведених категорій Інтернету речей полягає в галузі застосування. Споживчий IoT включає будь-які пристрої, які мають автоматизувати повсякденні завдання або керувати віддалено. Найчастіше його застосовують у побутових, домашніх умовах.

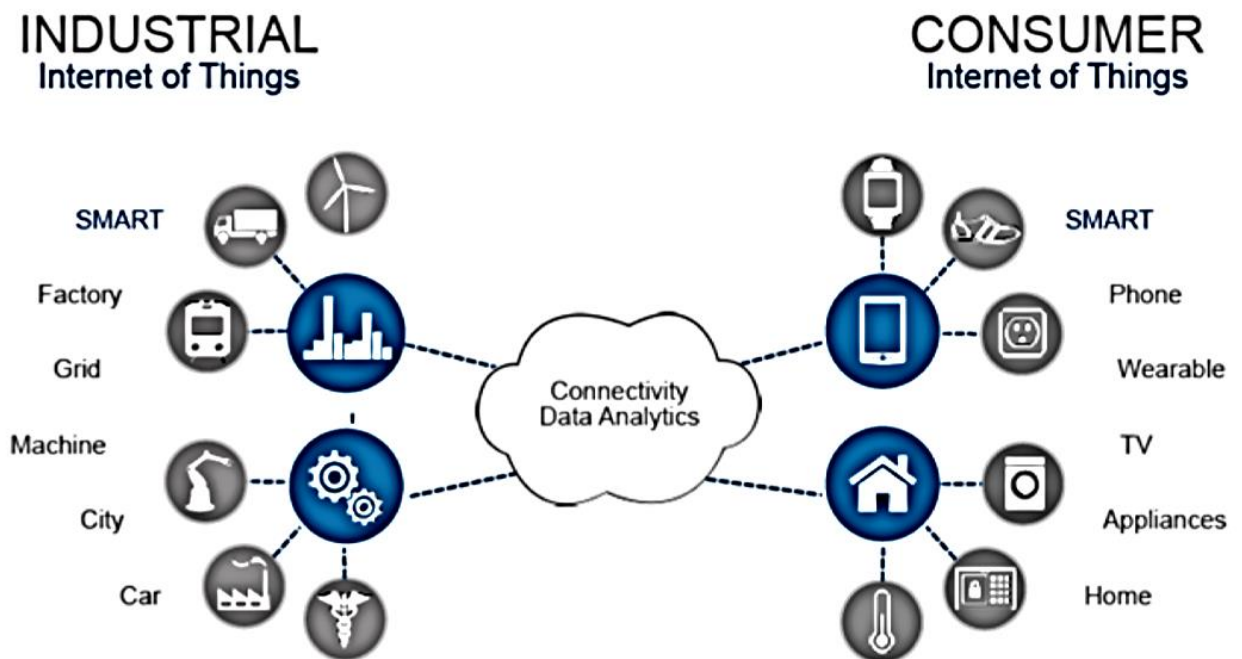


Рисунок 1.5 – Загальні категорії Інтернету речей

Споживчий Інтернет речей можна охарактеризувати як звичайну мережу. Це класичний IoT. Сюди включені побутові прилади, які поєднуються між собою. Застосовується така мережа у побуті. Споживчий IoT автоматизує рутинні завдання. Він дозволяє, наприклад, дистанційно керувати побутовими приладами, до цієї категорії можна зарахувати:

- розумні побутові прилади: холодильники, чайники,пилососи, лампи, розетки тощо;
- системи безпеки в будинку: електронні замки, камери спостереження із системою оповіщення та датчиками руху;
- системи розумного будинку, що відповідають за контроль витрати

води та електрики, включають та відключають опалення, стежать за можливими витокami (вода, газ);

- колонки та станції, що використовуються для об'єднання інших IoT-пристроїв, управління відповідним устаткуванням здійснюється за допомогою голосових команд;

- медичне обладнання, сюди можна віднести, наприклад, браслети, які відстежують пульс із подальшим надсиланням інформації лікарю;

За рахунок споживчого (класичного) IoT можна полегшити життя людини. Розумний будинок заощадить електрику та простежить за тим, щоб у приміщенні не сталася пожежа. Розумна камера попередить, якщо на житло буде скоєно напад або хтось спробує до нього вдертися. А розумні побутові прилади приготують їжу та забезпечать гарячий чай до повернення господаря.

Промисловий Інтернет речей (Industrial Internet of Things, IIoT) – зовсім інше поняття. Воно не застосовується у побуті, а зустрічається тільки на виробництві, складах, заводах, компаніях, лабораторіях тощо. Ключовим завданням IIoT є автоматизація та спрощення виробничих та подібних до них процесів, до цієї категорії можна зарахувати такі приклади [9]:

- датчики на устаткуванні та верстатах, які збирають дані про їхню роботу та допомагають запобігати поломкам;

- системи клімат-контролю, які аналізують температуру та вологість, автоматично регулюють мікроклімат у цеху чи на складі;

- датчики на продукції та деталях, які допомагають в інвентаризації та виявленні браку;

- системи дистанційного керування виробничим обладнанням, наприклад, системи дистанційного запуску верстатів;

- датчики забруднення, які аналізують викиди підприємства та стежать за дотриманням екологічних норм;

- "маячки" на транспорті, які дозволяють відстежувати вантаж у реальному часі;

– датчики на системах автомобіля, що допомагають відстежувати споживання бензину, зношування двигуна, температуру в рефрижераторі та інші параметри.

ІоТ потрібен, щоб знизити витрати на виробництві та уникнути збитків. Датчики на устаткуванні допоможуть зрозуміти, коли наступить час провести техобслуговування. Система клімат-контролю простежить, щоб товар не зіпсувався у спеку. Маячки на транспорті дозволяють зрозуміти, коли буде доставлений вантаж.

Дані, які збирають ці датчики і маячки, можна завантажувати в системи аналітики, щоб будувати прогнози і аналізувати роботу всього виробництва.

Приклад функціонування системи ІоТ наведений на рисунок 1.6 [10].

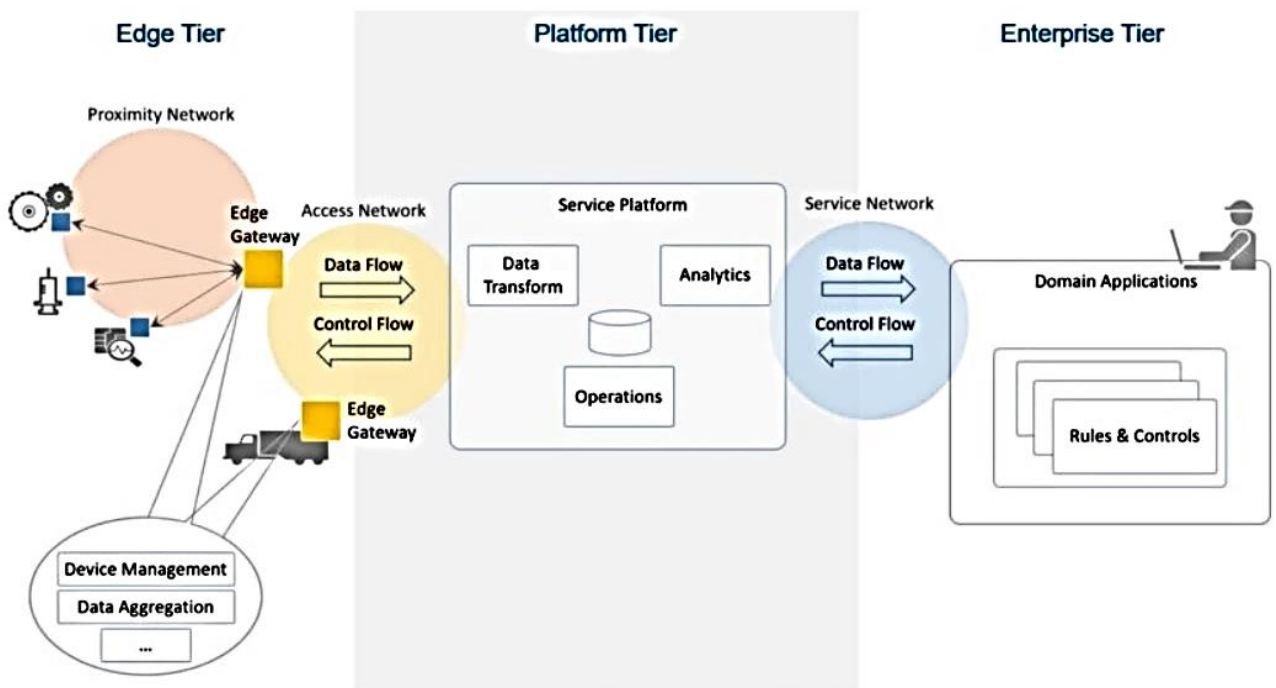


Рисунок 1.6 – Приклад функціонування системи ІоТ

Аналіз найкращих світових практик впровадження ІоТ показує, що основними сферами застосування рішень у сфері промислового Інтернету речей є виробництва, що характеризуються наявністю однієї чи кількох таких важливих умов:

– випуск широкої номенклатури продукції, використання значного

списку комплектуючих;

- потреба у підвищенні якості продукції та зниженні ступеня браку;
- потреба у забезпеченні ефективного сервісного обслуговування раніше поставленої продукції;
- потреба у зниженні експлуатаційних витрат виробництва;
- значна енергоємність виробництва;
- складні виробничі умови;
- потреба у забезпеченні високої продуктивності персоналу;
- потреба в оперативній діагностиці несправностей для технологічного устаткування зниження незапланованих зупинок виробництва;
- потреба у забезпеченні безпеки персоналу;
- необхідність системної інтеграції широкого спектра.

Розглянемо, як працює промисловий Інтернет речей на виробництві. Принцип роботи технології полягає в наступному: спочатку встановлюються датчики, виконавчі механізми, контролери та людино-машинні інтерфейси на ключові частини обладнання, після чого здійснюється збір інформації, яка згодом дозволяє компанії придбати об'єктивні та точні дані про стан підприємства. Оброблені дані доставляють у всі відділи підприємства, що допомагає налагодити взаємодію між співробітниками різних підрозділів та приймати обґрунтовані рішення. Отримана інформація може бути використана для запобігання позаплановим простоям, поломкам обладнання, скороченню позапланового техобслуговування та збоїв в управлінні ланцюжками постачання, тим самим дозволяючи підприємству функціонувати більш ефективно. При обробці величезного масиву неструктурованих даних їхня фільтрація та адекватна інтерпретація є пріоритетним завданням для підприємств. У даному контексті особливої значущості набуває коректне подання інформації у зрозумілому користувачеві вигляді, для чого сьогодні на ринку представлені передові аналітичні платформи, призначені для збору, зберігання та аналізу даних про технологічні процеси та події в реальному часі.

Промисловий Інтернет речей сприяє важливим якісним змінам в економіці:

- дані, які раніше були не доступні, зі зростанням проникнення вбудованих пристроїв є цінною інформацією про характер використання продукту та обладнання для всіх учасників виробничого циклу, є основним формуванням нових бізнес-моделей і забезпечують додатковий дохід від пропозиції нових послуг;

- віртуалізація виробничих функцій супроводжується формуванням «економіки спільного використання» (shared economy), що характеризується значно більш високою ефективністю та продуктивністю за рахунок підвищення використання наявних ресурсів, зміни функціоналу пристроїв без внесення змін до фізичних об'єктів шляхом зміни технологій управління ними;

- моделювання технологічних процесів, наскрізне проектування і, як результат, оптимізація ланцюжка створення вартості на всіх етапах життєвого циклу продукту в режимі реального часу, дозволяють виробляти штучний або дрібносерійний продукт за мінімальною ціною для замовника та з прибутком для виробника, що у традиційному виробництві можливе лише при масовому виробництві;

- еталонна архітектура, стандартизовані мережі та модель оренди замість оплати повної вартості володіння, роблять спільну виробничу інфраструктуру доступною для середнього та малого бізнесу, що полегшує їх зусилля з управління виробництвом, дозволяє прискорити реагування на вимоги ринку, що змінюються, і скорочення життєвого циклу продукції, і тягне за собою розробку та появу нових застосунків та сервісів;

- аналіз даних про користувача, його виробничі об'єкти (машини, будівлі, обладнання) та характер споживання відкривають можливості для постачальника послуги з покращення клієнтського досвіду, створення більшої зручності користування, кращого рішення та скорочення витрат клієнта, що веде до підвищення задоволеності та лояльності від роботи з цим

постачальником;

– функціонування різних галузей економіки безперервно ускладнюється під впливом розвитку технологій і все більше здійснюватиметься за рахунок автоматичного прийняття рішень на основі аналізу великих даних із підключених пристроїв, що призведе до поступового зниження ролі виробничого персоналу, у тому числі кваліфікованого, буде потрібна якісна професійна освіта, включаючи інженерну, спеціальні навчальні програми для працівників та тренінги.

Для подальшого аналізу необхідно з'ясувати різницю між споживчим і промисловим Інтернетом речей.

### 1.3 Порівняння споживчого і промислового Інтернету речей

Розберемо, чим споживчий IoT відрізняється від промислового Інтернету речей. Здається, що споживчий та промисловий інтернет речей не відрізняються: просто в одному випадку датчик температури стоїть на обігрівачі, а в іншому – на верстаті. Насправді, датчики і технології, які використовують для споживчого IoT (надалі – звичайного IoT, або IoT) і IIoT, мають суттєві відмінності, які детально розглянемо далі.

Споживчий Інтернет речей. Якщо чайник не вдасться увімкнути віддалено, користувачеві доведеться включити його вручну – це не так комфортно, але не надто страшно. Вдома набагато менше критичних завдань і малоімовірне виникнення ситуації, у якій поломка призведе до катастрофи. Ціна помилки при побутовому використанні невисока, тому надійність у IoT хоч і важлива, але не така критична, як у IIoT.

Промисловий Інтернет речей. Якщо не спрацює датчик на виробництві, то бізнес може втратити мільйони: може зламатися верстат, зіпсуватися товар на складі, загубитися цінний вантаж. Тому в промислових IoT-платформах насамперед приділяють увагу саме надійності: розробляють стійкі до відмови системи, дублюють важливі датчики, встановлюють резервні джерела

живлення на випадок відключення електроенергії.

Можна заперечити, що через поломку домашніх розумних пристроїв людина може затопити сусідів або втратити майно у пожежі. Але навіть такі втрати непорівнянні з катастрофами, які можуть спричинити порушення роботи IoT-системи: збитками у мільярди гривень або екологічними катастрофами.

Споживчий Інтернет речей. Його IoT-пристрої вкрай зрідка використовують складні алгоритми, зазвичай вони реагують на команди та найпростіші «подразники» на кшталт підвищення температури. А платформи, які їх поєднують, зазвичай просто віддають команди – складна аналітика даних у побуті потрібна доволі зрідка. Бувають і винятки, наприклад, розумні колонки, які вміють підтримувати діалог. Але навіть їм далеко до складних систем штучного інтелекту, аналітики та машинного навчання, які використовують у промисловості.

Промисловий Інтернет речей. Для завдань виробництва потрібні складніші рішення, щоб моделювати ланцюжок виробництва, виявляти ризики, збирати та аналізувати інформацію з безлічі різних датчиків. Для зберігання та аналізу інформації IoT-системи використовують складні технології: big data, штучний інтелект і алгоритми машинного навчання, так що їх якраз можна називати «розумними». Самі датчики в IoT теж часто бувають доволі складними: вони вміють аналізувати кілька різних фізичних показників та зіставляти їх між собою. Наприклад, датчик точки роси повинен уміти аналізувати тиск, вологість та температуру.

Споживчий Інтернет речей. Домашні «розумні» речі реагують на команди за кілька секунд, за побутових умов це допустимо. Якщо ви віддали команду закип'ятити чайник, то навряд чи помітите, чи включився він за одну чи три секунди — для домашніх справ це неістотно.

Промисловий Інтернет речей. У промисловості технологічні процеси часто вимагають реакції порядку мілісекунди. Датчики повинні постійно знімати актуальну інформацію, а пристрої миттєво реагувати на команди.

Отже питання швидкості обробки інформації для промислового Інтернету речей є вкрай важливими на відміну від споживчого IoT.

Споживчий Інтернет речей. Розумні домашні пристрої виходять до інтернету через звичайні канали. Інформація, яку вони передають, потребує лише базового захисту від вірусів та кіберзагроз. Навіть якщо станеться масова кібератака, катастрофу це навряд чи спричинить. Крім того, домашні пристрої зазвичай випускають великими партіями, і надлишковий захист дуже збільшить вартість пристроїв – їх ніхто не купуватиме.

Промисловий Інтернет речей. У бізнесі та промисловості ситуація набагато серйозніша – зловмисники можуть спробувати отримати доступ до «розумних» пристроїв підприємства, що загрожує мільйонними збитками, а іноді й промисловою катастрофою. Тому в мережах IoT приділяють підвищену увагу безпеці. Для захисту використовують такі методи:

- ізоляція від інтернету, робота тільки в мережі всередині офісу чи заводу;
- шифрування промислових даних, наприклад, використовуючи алгоритм AES-256; щоб його зламати, потрібно підібрати ключ довжиною 256 біт – навіть сучасному суперкомп'ютеру для цього потрібно більше часу, ніж передбачуваний вік нашого всесвіту;
- захист програмного забезпечення та операційних систем;
- системи детектування та запобігання вторгненням, які сповіщають, якщо хтось

Споживчий Інтернет речей. Домашні пристрої генерують зовсім невеликі обсяги інформації: у багатьох випадках протягом року її не накопичиться і кількох гігабайт. Для зберігання та обробки цієї інформації не потрібні об'ємні сховища та спеціальні складні інструменти.

Промисловий Інтернет речей. Датчики на устаткуванні можуть щодня генерувати гігабайти інформації. Все це потрібно швидко збирати, записувати до баз даних, зберігати і при необхідності витягувати для аналізу. Для цього підприємства будують цілі системи роботи з Big Data і

використовують спеціальні підходи до аналітики даних.

Часто домашні «розумні» пристрої зберігають дані прямо в собі або в пам'яті смартфона, з якого відбувається керування розумним будинком. Для ПоТ у багатьох випадках потрібні окремі сховища величезних обсягів. Часто для цього підприємства використовують хмари.

Споживчий Інтернет речей. Для побутового IoT існують готові рішення: набір з датчиків і платформи, що управляє, за допомогою якого можна швидко побудувати розумний будинок або зробити розумним свій автомобіль. Зазвичай з провадженням таких систем особливих проблем не виникає. У низці випадків споживач зможе самостійно, згідно з інструкцією, налагодити відповідну систему. У інших випадках практично завжди достатньо послуг спеціаліста – налагоджувальника системи.

Промисловий Інтернет речей. Для застосування ПоТ потрібне стабільне з'єднання. Важливо, щоб пристрої постійно були пов'язані між собою і з платформою, що управляє, тому що без цього нічого працювати не буде. В рамках одного підприємства це можна забезпечити за рахунок швидкої провідної мережі та середовища з мінімумом перешкод для передачі даних повітрям.

Датчики. Саме датчики будуть збирати інформацію про стан обладнання, мікроклімат чи інші показники, які важливо відстежувати в процесі роботи. Безпосередньо датчики нічого не аналізують і не автоматизують — вони тільки збирають інформацію.

Інструменти для збору даних, аналітики та прийняття рішень. Дані інструменти збирають інформацію з датчиків, обробляють її, аналізують, відображають результати аналізу. Такі рішення можуть бути розгорнуті на власних серверах компанії, а можуть працювати у хмарі. Для промислових завдань є ПоТ-рішення, наприклад, хмарні платформи Інтернету речей, які інтегрують у бізнес-процеси компанії. Таку платформу можна впровадити поверх існуючих рішень, наприклад, інтегрувати з системами SCADA або MES, базами даних підприємства та іншими інструментами, що вже

працюють. Хмарні технології роблять IoT доступнішим не тільки для великих, але і для малих та середніх підприємств. Для підключення хмарного рішення не потрібно закуповувати обладнання для зберігання та обробки даних, вибудовувати складну інфраструктуру та турбуватися про її обслуговування. Всі потрібні інструменти доступні в хмарі – можна просто підключити до них датчики або інші джерела даних, щоб зробити своє виробництво «розумнішим».

IoT тепер стає прикладним інструментом для будь-якого бізнесу. Хмарні рішення дозволяють будь-якій компанії впровадити IoT-технології, щоб збільшити прибуток і не відставати від конкурентів.

Трафік, що створюється датчиками споживчого Інтернету речей, не створює інформаційних потоків Big Data, такі потоки більш адекватно характеризуються дискретними процесами, ніж неперервними. Крім того, що створювані дані не потребують для зберігання великих обсягів зовнішньої пам'яті, створений ними трафік доволі легко прогнозується. Викиди трафіку є доволі рідкими, а затримки в обслуговуванні, що викликані ними, не вплинуть на якість функціонування системи споживчого Інтернету речей. Отже трафік для даного типу Інтернету речей можна вважати приблизно (умовно) стаціонарним, орієнтованим на фіксований набір обслуговуючих пристроїв.

Трафік промислового Інтернету речей формується багатьма інформаційними потоками, які надходять з різних груп кінцевих пристроїв, що надають різноманітну за характером та структурою інформацію. Датчики IoT в залежності від наданих функцій можуть надавати як дискретну, так і неперервну, наприклад, відео, різноманітну інформацію. Надходячи до обчислювальних пристроїв шару аналізу та обробки окремі інформаційні потоки інтегруються у єдиний агрегований потік даних, який набуває властивостей, кардинально відмінних від властивостей окремих потоків. На відміну від трафіка споживчого Інтернету речей трафік IoT стає нерівномірним та суттєво нестаціонарним, трапляється багато

непрогнозованих викидів. Це, в свою чергу, сприяє створенню черг до пристроїв аналізу, обробки та зберігання інформації, а, отже, до непередбачуваних затримок інформаційних пакетів ПоТ, що неприпустимо при нормальному функціонуванні системи.

Для вирішення проблеми при використанні хмарного шару ПоТ необхідно збільшувати сумарну потужність обчислювальних пристроїв центру обробки даних (ЦОД), що обслуговують трафік системи ПоТ. Для цього можна розглядати два підходи:

- статичний, при якому обчислювальні потужності надаються із розрахунку на максимальну інтенсивність трафіку ПоТ;
- динамічний, при якому обчислювальні ресурси виділяються по мірі потреби.

Перший варіант, звісно, є економічно недоцільним як для споживача послуги, так і для надавача послуги, тому що в такому разі обчислювальне обладнання буде незадіяне більшість з виділеного часу для системи ПоТ. Отже, більш доцільним є другий варіант. Але постійна зміна потужностей обчислювального ресурсу у хмарному середовищі є непродуктивною та вельми затратною операцією, для якої може бути потрібною тимчасова зупинка основного процесу, що є неприпустимим для систем ПоТ. Найбільш сприятливим способом реалізації динамічного варіанта для даного випадку є надання процесу властивості еластичності. На сьогодні вже багато ЦОД використовують гіперконвергентні платформи, які є еластичними за принципом формування як апаратної, так і програмної складової. Розширення та звуження віртуального середовища ПоТ відбувається доволі швидко, без зупинки основного процесу.

Але в будь-якому разі додавання додаткового обчислювального пристрою до еластичної системи займає деякий, хоч і невеликий час, що може негативно вплинути на якість функціонування системи ПоТ. Бажано було б розширення потужності зробити заздалегідь, щоб не сприяти створенню черги на аналіз, обробку та зберігання інформації, що надходить

до хмари із системи IoT. Тому виникає необхідність в розробці методу короткочасного прогнозування трафіку промислового Інтернету речей, що набуває еластичної властивості при обробці на гіперконвергентній хмарній платформі.

У ході проведеного аналізу були розглянуті можливості та завдання сучасних систем Інтернету речей. Зокрема, відмічена роль Інтернету речей у сучасному світі, наведені темпи його зростання та розширення сфери застосування. Також наведена класифікація Інтернету речей, що поділяє його на дві великі категорії – споживчий Інтернет речей і промисловий (індустріальний) Інтернет речей, головна відмінність між якими полягає в галузі застосування.

Проведено порівняння споживчого і промислового Інтернету речей, зокрема по таким характеристикам, як надійність та ціна помилки, складність алгоритмів, швидкість обробки інформації, безпека, обсяги даних, що збираються, схеми впровадження та особливості трафіку.

Обґрунтовані необхідність динамічного підходу до виділення еластичних хмарних обчислювальних ресурсів системам промислового Інтернету речей та необхідність в розробці методу короткочасного прогнозування трафіку промислового Інтернету речей, що набуває еластичної властивості.

## 2 ПРИНЦИПИ ОРГАНІЗАЦІЇ ТА ВЛАСТИВОСТІ ЕЛАСТИЧНИХ СИСТЕМ

### 2.1 Концепція еластичної системи

Обчислювальні ресурси, на яких здійснюються обчислення різної ступеня складності, зазнали значні зміни за останні кілька десятиліть років, так само як і кількість людських трудовитрат на виконання даних обчислень. Раніше програмісту потрібно виносити програми на перфокарти, після чого завантажувати їх в обчислювальну машину. Самі обчислювальні машини займали цілі зали і могли виконувати тільки одну програму в конкретний момент часу. Великим проривом в організації обчислювального процесу стала поява можливості паралельного виконання кількох програм, що дозволяло масштабувати обчислення на одній конкретній машині. Управління ресурсами для паралельного виконання обчислювальних програм здійснювала загальна управляюча програма операційної системи ЕОМ – супервізор, яка монополізувала всі доступні ресурси та розподіляла їх між завданнями. Але залишалася одна велика проблема – територіальна прив'язаність програміста або користувача програмного забезпечення до конкретної обчислювальної машини. Для рішення даного питання була запропонована ідея об'єднання комп'ютерів в деяку структуру, названу в результаті комп'ютерної мережею. Далі з'явилася глобальна мережа Інтернет. З появою Інтернету та можливості публічного доступу до конкретного програмного забезпечення також зросли вимоги до апаратної складникової інформаційно-обчислювальних систем, виникла необхідність збільшення обчислювальної потужності відповідно зростаючому навантаженню.

Спочатку збільшення обчислювальної потужності досягалося тільки шляхом заміни комплектуючих більш продуктивними. Таке масштабування було названо вертикальним і в принципі використовується і по цей день. Однак також виникала ситуація, коли програмне забезпечення не використовувало всіх доступних обчислювальних ресурсів, що призводило

до їх простою та додаткового споживання електроенергії. ІІ. І свого часу було поставлено питання зниження простою обчислювальних ресурсів. Даний питання був вирішено з допомогою створення нового рівня абстракції, котрий був названий гіпервізором. Гіпервізор за своєю суттю є супервізором над супервізорами, що контролює їх роботу та ізоляцію один від одного. Дана технологія дозволила розміщувати на одному обчислювальному вузлі більше однієї операційної системи, що відкрило нові можливості щодо організації обчислювального процесу і розміщення програмних засобів на обчислювальних машинах. Запуск декількох екземплярів операційних систем на одному обчислювальному вузлі забезпечив можливість іншого типу масштабування, названого горизонтальним, тобто. коли для нарощування продуктивності запускається черговий екземпляр програми.

Згодом стало зрозуміло, що зростання потреб у обчислювальних ресурсах вимагає іншого підходу, відмінного від того, в парадигмі якого сервери розміщувалися локально у компаніях або науково-технічних та освітніх організаціях. З'явилися спеціальні відведені місця та споруди, названі центрами обробки даних (ЦОД), де розміщені основні серверні обчислювальні ресурси, існуючі на даний час. Центри обробки даних виконують важливу функцію – беруть на себе відповідальність за розміщення і безперебійне енергопостачання серверного обладнання, а також його мережнудоступність. Гіпервізор ЦОД дозволяє запускати екземпляри операційних систем, названі віртуальними машинами, на одному або кількох обчислювальних вузлах. Таким чином з'явилося поняття віртуалізованої інфраструктури, коли обчислення виконуються на віртуальних машинах без прив'язки до конкретного фізичного сервера. Такі обчислення разом із можливістю мережного доступу були названі хмарним, так як система представляється на рівні віртуалізації, а мережна та обчислювальна конфігурації реального обладнання невідомі і приховані від користувача подібно хмарі [10].

Даний клас систем надає інтерфейс управління віртуалізованими

обчислювальними і мережними ресурсами гіпервізора. В даний час все більшу популярність набирає віртуалізація на рівні операційної системи, тобто ядро операційної системи виступає як гіпервізор. Така віртуалізація називається контейнерною. Контейнери поділяють між собою одне ядро операційної системи хоста (рисунок 2.1).



Рисунок 2.1 – Рівні віртуалізації обчислювального процесу

Є кілька реалізацій у різних операційних системах. На Solaris цей механізм називається зонами, в FreeBSD – це jail, у Linux – це механізм на основі cgroups та namespaces. Контейнерна віртуалізація надає черговий рівень абстракції, що дозволяє ізолювати різні оточення всередині однієї операційної системи. У процесах розробки програмного забезпечення перед програмістами часто стоять завдання по роботі з різними версіями бібліотек і залежностей. Для того, щоб протестувати роботу якоїсь програми з різними залежностями, у випадку віртуальних машин, програмісту необхідно повністю запустити черговий екземпляр операційної системи, що зазвичай є досить трудозатратним для програміста і знижує його продуктивність. У випадку з контейнерами процес тестування та розробки прискорюється, оскільки запуск контейнера несе в собі менші накладні витрати, ніж запуск віртуальної машини з повноцінним образом операційної системи. Подібно

гіпервізору і віртуальним машинам в випадку контейнерів також були розроблені системи, полегшують управління контейнерними оточеннями. Такі системи називаються системами контейнерної кластеризації, найбільш популярними з яких є Kubernetes та Docker Swarm. Системи контейнерної кластеризації без проблем працюють поверх хмарних платформ, забезпечуючи тим самим широкий набір можливостей з управління обчислювальними ресурсами та розробці і супроводу програмного забезпечення.

На сьогодні широке поширення отримала організація обчислень за допомогою різних технологій віртуалізації. Такий підхід дозволяє відійти від класичного розуміння обчислювального вузла як деякої апаратної одиниці, жорстко прив'язаної до конкретної фізичної інфраструктури, надаючи тим самим широкий набір можливостей по управлінню доступними обчислювальними потужностями. З часом утворилося два класи систем, що використовують віртуалізацію як основу побудови обчислювальної інфраструктури – це хмарні системи та системи контейнерної кластеризації. Хмарні системи надають користувачеві можливість побудови власної віртуальної інфраструктури поверх фізичної інфраструктури центру обробки даних. наприклад: створювати і видаляти обчислювальні вузли, мережні ресурси і системи зберігання даних. Обчислювальні вузли можуть створюватися на різних серверах та переміщатися між ними, забезпечуючи при цьому відсутність відмов в обслуговування користувальницьких запитів, що підвищує доступність і відмовостійкість рішень, розроблених на базі хмарних систем. Однак в останні кілька років широке поширення також отримують системи контейнерної кластеризації, що представляють обчислювальні вузли у вигляді деяких «контейнерів» [11]. Дані системи використовують легковагову віртуалізацію, що дозволяє запускати декілька ізольованих один від одного «контейнерів» усередині однієї операційної системи, підвищуючи тим самим ефективність використання обчислювальних ресурсів та знижуючи витрати на серверне обладнання та

ліцензії.

Найбільш поширеним в даний час є комбінований варіант розгортання систем контейнерної кластеризації та хмарних систем із залученням переваг обох рішень.

Однією із основних переваг як хмарних систем, так і систем контейнерної кластеризації є можливість горизонтального масштабування обчислювальних ресурсів відповідно до навантаження, створюваного безпосередніми користувачами запущених у системі застосунків. Тому термін «еластична система» був введений для систем, що володіють властивістю горизонтального масштабування.

## 2.2 Способи реалізації еластичних систем

Еластичну систему можна представити як низку однотипних обчислювальних вузлів, навантаження між якими розподіляється за допомогою балансувальника навантаження.

Балансувальник навантаження приймає вхідні запити в систему і на основі закладених в нього алгоритмів розподіляє запити між обчислювальними вузлами системи. Додавання або видалення обчислювальних вузлів у систему дозволяє адаптувати її сумарну продуктивність під конкретну інтенсивність вхідного потоку запитів. Усі зміни в кількості вузлів реєструються балансувальником навантаження для подальшого коректного спрямування запитів тільки до існуючих справних вузлів. На балансувальник навантаження також лягає завдання по запобіганню обробки запиту на несправному обчислювальному вузлі. Даний механізм заснований на пробах життєздатності вузлів, що полягають у періодичному опитуванні того чи іншого вузла на предмет його працездатності по заздалегідь визначеним критеріям.

Еластичні системи можуть бути збудовані на різних рівнях абстракції з точки зору віртуальних машин. Так як основною властивістю таких систем є саме можливість горизонтального масштабування, тобто додавання або видалення однотипних вузлів з обчислювального процесу, то, беручи до уваги можливість сучасних програмно-апаратних засобів обчислювальної техніки, будувати такі системи можна, можливо на різних рівнях моделі OSI. Наприклад, на рівні серверного апаратного забезпечення як обчислювальні вузли можуть виступати серверні конфігурації різних виробників, які мають мережевий інтерфейс та встановлену операційну систему. Як балансувальники навантаження можуть виступати різні мережні пристрої з можливістю балансування трафіку на каналному та/або мережному рівнях або спеціально призначені для таких цілей апаратні балансувальники (рисунок 2.2).

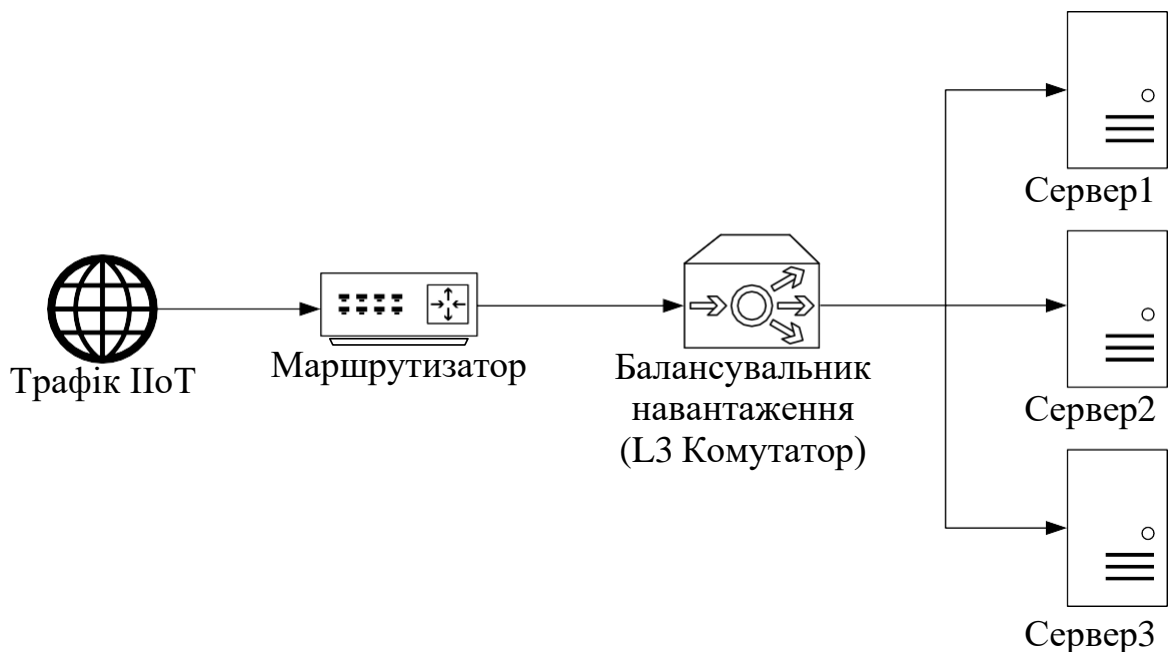


Рисунок 2.2 – Еластичний трафік ЦОД

Додавання і видалення обчислювальних вузлів в таких конфігураціях здійснюється за допомоги протоколів комутації і маршрутизації з балансуванням трафіку і різних інтерфейсів управління серверними платформами.

На наступному рівні абстракції еластичні системи будуються за допомогою різних технологій програмної віртуалізації на основі гіпервізорів, що дозволяють запускати поверх себе кілька однотипних екземплярів гостьових операційних систем (ОС), що називаються віртуальними машинами (ВМ), емулюючи цим апаратне забезпечення. Гіпервізори першого типу працюють поверх серверних конфігурацій і запускаються прямо на фізичному апаратному забезпеченні, будучи тим самим для гостьових ОС безпосередньо хостовою ОС. Гіпервізори другого типу запускаються поверх хостової ОС. На основі гіпервізорів обох типів працюють хмарні платформи, що дозволяють гнучко керувати віртуальними машинами та їх ресурсами залежно від тих чи інших умов.

До першого типу гіпервізорів можна віднести продукт компанії VMWare під назвою ESXi, на його основі розроблено пропрієтарна хмарна платформа VMWare vSphere.

До другому типу відноситься, наприклад, технологія на основі ядра Linux під назвою KVM, на її основі функціонує, наприклад, хмарна платформа з відкритим вихідним кодом OpenNebula. На описаному рівні абстракції в якості обчислювальних вузлів еластичною системи виступають віртуальні машини з гостьовими ОС (рисунок 2.3).

У якості балансувальника навантаження в даному випадку можуть виступати різні програмні засоби, наприклад, програмні реверс-прокси, такі як HAProxy або Nginx, або програмні балансувальники, що входять до постачання хмарної платформи. Мережна зв'язність в даному випадку забезпечується за допомогою віртуальних мереж, конфігурованих через інтерфейс гіпервізора.

Самим високорівневим рішенням для побудови еластичних систем на даний момент є технології контейнеризації та контейнерної оркестрації. Саме поняття контейнера в обчислювальній техніці з'явилося досить давно, починаючи з таких технологій як Solaris Zones, FreeBSD jail і Linux lxc.

Контейнер – це ізольований процес на рівні ядра операційної системи, який за допомогою механізмів ізоляції отримує доступ до обчислювальних ресурсів основної ОС. Таким чином, головною відмінністю контейнера від віртуальної машини є його рівень ізоляції від хостової ОС. Контейнер використовує той самий екземпляр ядра, що і хостова ОС. Віртуальна машина ж інкапсулює в собі повністю самостійне ядро. В даний час промисловим стандартом у технологіях контейнеризації є програмний продукт Docker, заснований на механізмах ізоляції ядра Linux.

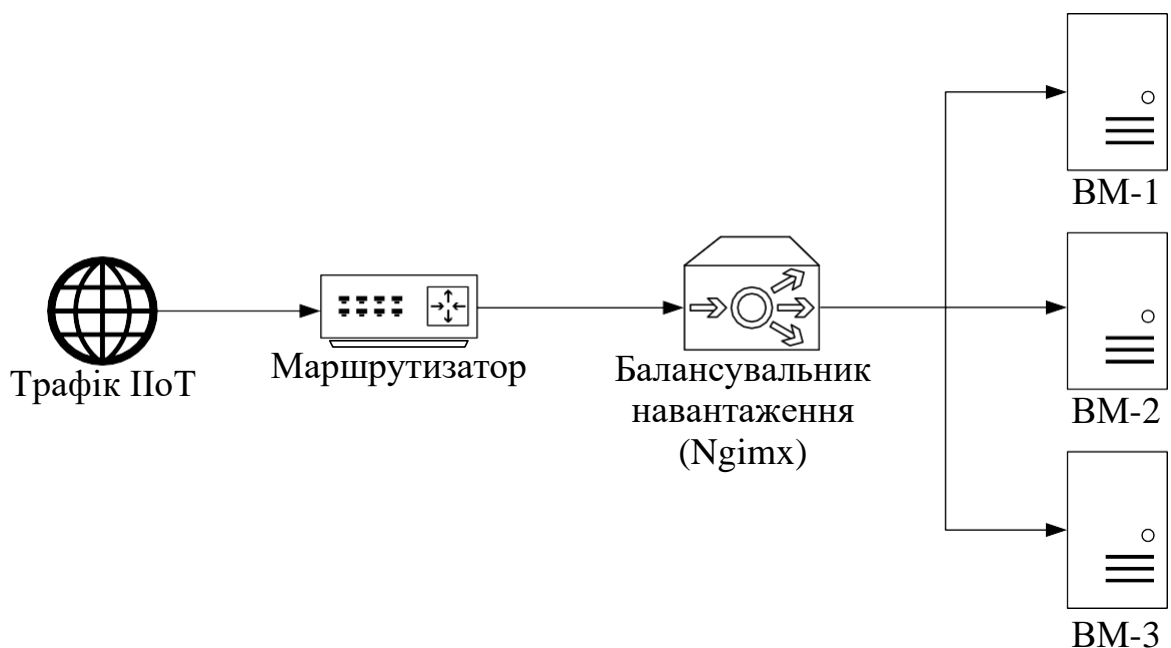


Рисунок 2.3 – Еластичний трафік хмарної платформи

Для організації балансування навантаження та мережної зв'язності контейнерів були розроблені системи контейнерної кластеризації або системи. Дані системи призначені для управління життєвим циклом контейнерів, їх розміщення на хостових ОС і забезпечення відмовостійкості контейнерних інсталяцій в цілому (рисунок 2.4).

Найбільш популярною системою контейнерної кластеризації на цього-сьогоднішній день є Kubernetes.

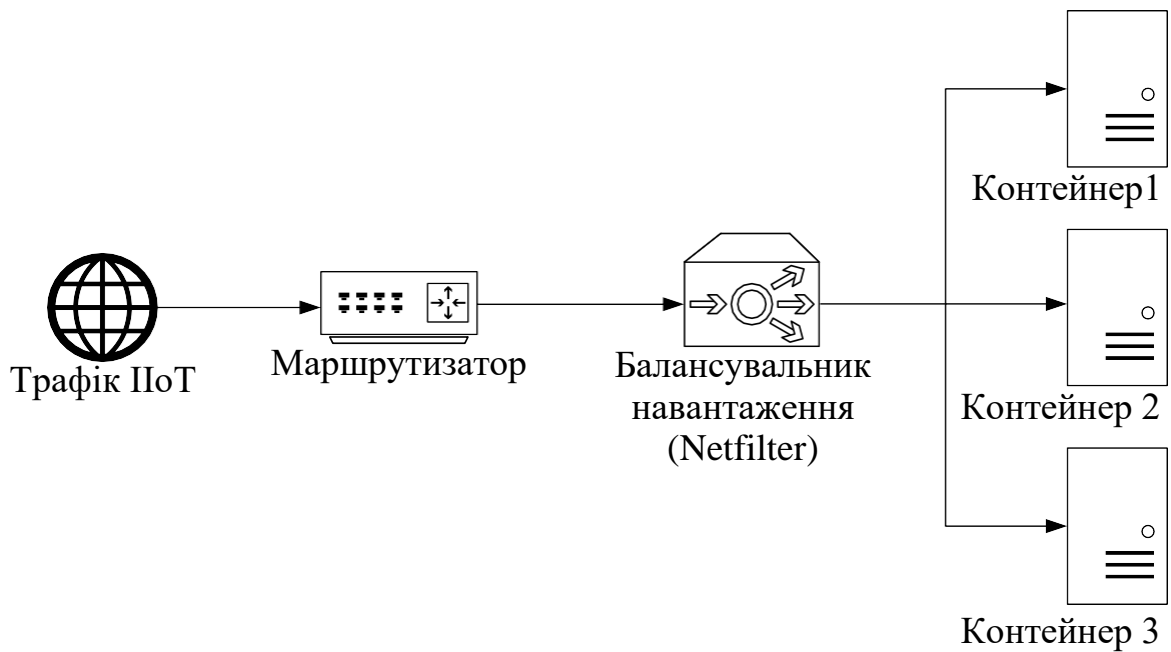


Рисунок 2.4 – Еластичний трафік контейнерної оркестрації

Таким чином, на даному рівні абстракції обчислювальним вузлом еластичної системи є саме контейнер. Як балансувальник навантаження в даному випадку може виступати як вбудований у систему контейнерної кластеризації балансувальник на основі правил netfilter, так і програмні реверс-проксі в сукупності з налаштованими системами правилами netfilter.

Усе більше сучасних інсталяцій еластичних систем працюють на рівні абстракції, створеному технологіями контейнеризації. Але дані інсталяції систем можна розглядати як з погляду логічною організацією, так і фізичною. Під логічною організацією розуміється побудова системи саме з погляду виконання поставлених перед нею завдань. У разі рівня технології контейнеризації логічна організація включає в себе контейнери як обчислювальні вузли і засоби системи контейнерної кластеризації як балансувальник навантаження. Під фізичною ж організацією розуміється виділення обчислювальних потужностей, починаючи з фізичних серверів. Таким чином, фізична організація сучасних еластичних систем включає всі вищеописані рівні абстракції. Наприклад, обчислювальний вузол являє собою контейнер усередині певної ОС. Дана ОС є гостьовою ОС під управлінням конкретного гіпервізора. Гіпервізор запускається або поверх

хостової ОС, або поверх фізичного сервера. Рівні абстракції у випадку з балансувальником навантаження розташовуються в аналогічному порядку. Схожу багаторівневу структуру пропонують в якості сервісу для організації еластичних систем більшість популярних хмарних провайдерів. Як приклади таких сервісів можна навести Amazon Elastic Kubernetes Service, Google Kubernetes Engine та Azure Kubernetes Service.

### 2.3 Управління обчислювальними ресурсами еластичних систем

Сучасні системи працюють з різними потоками вхідних запитів. Ці потоки можуть надходити як від звичайних користувачів, так і від інших систем різного призначення. Зі зростанням чи зниженням кількості користувачів або суміжних систем, що посилають у конкретний момент часу запити на обробку цільової системи, відбувається також зміна інтенсивності вхідного потоку запитів. Різна інтенсивність створює різне навантаження на обчислювальні ресурси еластичної системи. Процес підвищення або зниження продуктивності системи для обробки змінного потоку вхідних запитів до системи називається масштабуванням.

До появи різних платформ віртуалізації і контейнерізації масштабування інформаційно-обчислювальних систем здійснювалося вручну оператором системи. Даний процес займав довгий час, тому що доводилося здійснювати велику кількість операцій ручного конфігурування обчислювальних вузлів і їх мережної зв'язності з балансувальником навантаження. Це впливало не тільки на вартість обслуговування системи, але й на своєчасність реакції системи на зміни у навантажувальних процесах. Хмарні платформи та платформи контейнерної кластеризації істотно прискорили процес зміни кількості обчислювальних ресурсів, доступних системі, шляхом забезпечення можливості їх горизонтального масштабування в автоматичному режимі. Цей підхід був названий автоматичним масштабуванням. У розглядаємих платформах за реалізацією цього підходу в

більшості випадків відповідає окремий сервіс автоматичного масштабування. Приклад типового алгоритму автоматичного масштабування, заснований на граничних значеннях миттєвого завантаження вузлів, наведений на рисунку 2.5.

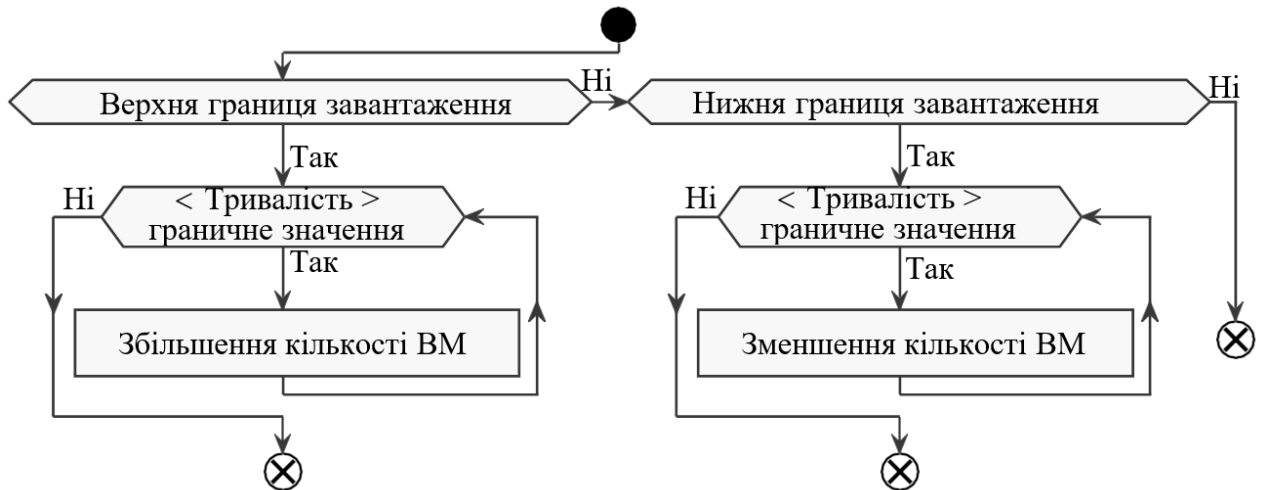


Рисунок 2.5 – Типовий алгоритм автоматичного масштабування

При реалізації програмних систем на основі хмарних платформ та платформ контейнерної оркестрації перед сервісом автоматичного масштабування ставиться наступний ряд завдань, які розглянемо більш детально.

Перевантаження в типових системах обумовлені тим, що інтенсивність надходячих запитів вище, ніж продуктивність системи. Система, функціонуюча на основі клієнт-серверний моделі, на запити користувача відправляє програмно визначені відповіді. Час відповіді формується на основі сукупності факторів. Першим з них є час поширення пакетів мережею. Другим, не менш важливим фактором, є час обслуговування запиту у системі. Користувач хмарної системи зазвичай чекає відповіді в рамках певного часового інтервалу. Якщо говорити про великі хмарні системи та застосунки, розміщені у них, час відповіді або відгуку системи встановлюється в рамках вимог користувача.

При проектуванні системи та її введенні в експлуатацію закладається

деякий запас продуктивності, щоб система справлялася з навантаження на старті. Однак згодом, наприклад, через зростання популярності програми, розміщеної на хмарній платформі, навантаження на систему може значно зрости. В якийсь момент навантаження перевищить поріг, при якому продуктивності системи достатньо для функціонування без перевантажень. Такий режим характеризується повною завантаженістю обчислювальних ресурсів на аналізованому проміжку часу. Все це призводить до погіршення характеристик функціонування системи. Зростають черги запитів, що у результаті безпосередньо впливає на час відповіді самої системи на запити користувача. Таким чином, домовленості не виконуються, та користувачі залишаються незадоволені якістю обслуговування, що безпосередньо наводить до фінансових втрат для бізнесу. Оперативне реагування на підвищення навантаження на систему та додавання ресурсів є важливим завданням, яке ставлять перед сервісами автоматичного масштабування сучасних хмарних платформ. Наявність таких можливостей дозволяє системі вчасно адаптуватися до зростаючого навантаження і зберегти якість надання послуг.

У контексті хмарних обчислень можна, можливо виділити різні рівні з точки зору власників обчислювальних ресурсів або їх орендарів. На рівні ЦОД власником фізичних обчислювальних ресурсів у більшості випадків є сам власник конкретного ЦОД. Однак користувачі можуть скористатися послугою розміщення власного обладнання в ЦОД, сплачуючи оренду місця для розміщення, а також прилеглої інфраструктури (мережне обладнання, електроенергія, охолодження тощо). Далі на фізичному обладнанні вже можуть бути розгорнуті власні платформи віртуалізації для організації хмарних обчислень. У даному випадку, власником хмарної системи буде безпосередньо ственно той, хто розгорнув цю платформу поверх фізичного обладнання ЦОД. Користувачем хмарної системи буде той, хто орендує віртуалізовані потужності всередині даної платформи. Якщо розглядати платформи контейнерної кластеризації, то в контексті хмарних обчислень

користувач також повинен мати хостові машини для розміщення контейнерів, незважаючи на те, що обслуговування і розміщення керуючих вузлів кластера бере на себе власник хмарної платформи. У даному випадку користувач платформи контейнерної кластеризації також є користувачем хмарної платформи.

Таким чином, оцінка вартості володіння обчислювальними ресурсами визначається виходячи із рівня абстракції організації обчислювального процесу. Для власника ЦОД – це витрати на обслуговування приміщення, електроенергію, обладнання і т.д. Для власника хмарної платформи це можуть бути ті ж самі витрати, що і для власника ЦОД, якщо, наприклад, хмарна система надається власником ЦОД як послуга. Але власник хмарної платформи може розгорнути її поверх орендованого у власника ЦОД обладнання, тим самим знижуючи витрати на вартість оренди цього обладнання та будучи таким чином користувачем послуг власника ЦОД. Для користувача хмарної платформи витрати полягають в оренді віртуалізованих обчислювальних ресурсів у власника хмарної платформи у разі, якщо він сам не є власником цієї хмарної платформи (рисунок 2.6). А саме: віртуальних машин, балансувальників навантаження- ки, систем зберігання даних тощо.



Рисунок 2.6 – Зони відповідальності в хмарному середовищі

Для користувача платформи контейнерної кластеризації до витрат додаються послуги з розміщення і обслуговування керуючих вузлів кластера в хмарній платформі у випадку, якщо користувач не є власником хмарної платформи і не розгортає платформу контейнерної кластеризації самостійно.

Широке поширення еластичних систем очікувано привело до різноманітності сфер їх застосування. На базі хмарних платформ та платформ контейнерної кластеризації в даний час розробляються застосунки найширшого спектра цільових призначень, починаючи з платформ Інтернету речей, що обробляють інформацію з датчиків та вимірювальних пристроїв та закінчуючи автоматизацією високорівневих бізнес-процесів великих корпорацій. Така різноманітність функціональних призначень означає також і різноманітність навантажувальних процесів, які можуть відчувати системи в залежності від цільових функцій. Сервіси автоматичного масштабування сучасних платформ спрямовані на аналіз миттєвої поточної завантаженості обчислювальних ресурсів системи, таких як процесорний час та оперативна пам'ять. Таким чином, у даних реалізаціях відсутній який-небудь аналіз користувальницького навантаження на систему. Наявність можливості такого аналізу в сервісах автоматичного масштабування еластичних систем дозволить завчасно визначати поведінку навантажувального процесу і проактивно реагувати на його зміни, адаптуючи при цьому структуру самої системи (рисунок 2.7).

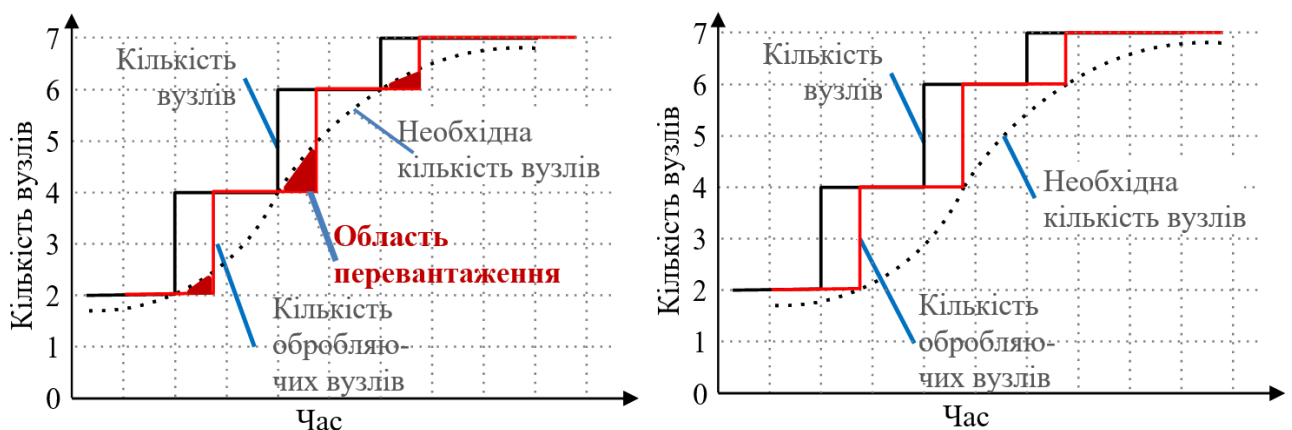


Рисунок 2.7 – Проактивне масштабування

Отже, для запобігання виникненню області перевантаження необхідно завчасно, не пізніше ніж на величину інтервалу часу підключення, зробити замовлення на перемасштабування, а це можливо тільки при наявності короткочасного прогнозу величини трафіку, що надходить до хмари.

Одна і та ж структура системи не може бути використана для обробки навантаження будь-якого характеру. Якщо навантаження на систему значно зростає і її продуктивності перестає вистачати, то система переходить в перевантажений стан, що веде до погіршення якості обслуговування користувачів системи. В той же час, якщо навантаження на систему в якийсь момент часу значно знизилось, то виникає простоювання обчислювальних ресурсів, так як запас продуктивності системи стає занадто великий. Це призводить до необґрунтованих витрат на ресурси, що не виконують корисної роботи. У широко представлених на ринку хмарних платформах визначення необхідного числа обчислювальних ресурсів задається оператором системи вручну. Основний недолік даного підходу полягає в тому, що оператор заздалегідь не знає як поведеться навантажувальний процес. Таким чином, може виникнути ситуація, коли зазначеною фіксованою «порцією» може бути замало або занадто багато, що в результаті призведе до перевищення або недовиділення необхідної кількості обчислювальних вузлів. Наявність можливості аналізу навантажувального процесу з точки зору необхідної кількості обчислювальних ресурсів в залежності від реального чи прогнозованого показника навантаження дозволить зменшити кількість помилок щодо перевиділення або недовиділення обчислювальних вузлів в еластичній системі.

Хостові вузли, на яких запускаються обчислювальні вузли системи, мають обмежені ресурси, такі як кількість фізичних або віртуальних процесорів, оперативної пам'яті або дискового простору. Отже, кожен хостовий вузол може розмістити у собі обмежену кількість обчислювальних вузлів. На хмарну платформу або платформу контейнерної кластеризації

покладаються завдання по розміщенню обчислювальних вузлів на хостових вузлах. За це відповідає планувальник, який є окремою структурною одиницею платформи (рисунок 2.8).

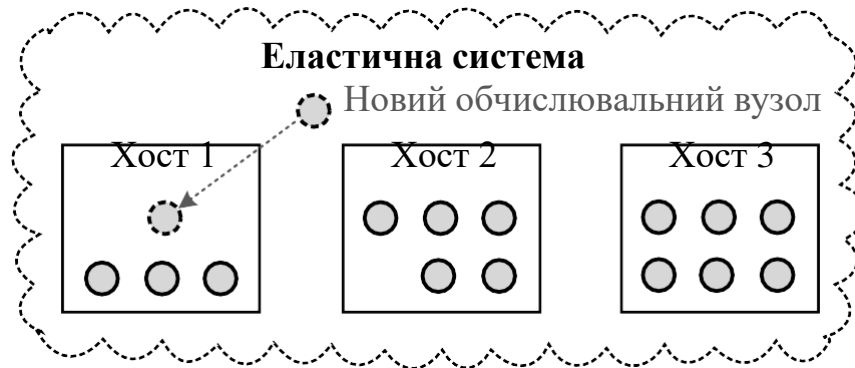


Рисунок 2.8 – Перерозподіл віртуальних машин планувальником

Планувальник вибирає найбільш підходящий хостовий вузол і резервує вільні обчислювальні ресурси для розміщення чергового обчислювального вузла. А у разі потреби відключення обчислювального вузла обирає підходящий хостовий вузол і вивільняє ресурси.

Якщо ресурсів хостових вузлів недостатньо для розміщення чергового обчислювального вузла, то планувальник повинен сповістити оператора про необхідність виділення додаткових обчислювальних ресурсів. Цей процес складний з погляду автоматизації з урахуванням алгоритмів, що використовуються в сучасних сервісах автоматичного масштабування. Дана проблема найбільш поширена при роботі одночасно з хмарними платформами та платформами контейнерної кластеризації, тому що остання стежить за кількістю вільних віртуальних процесорів на віртуальних машинах хмарної платформи і не дозволяє запустити черговий контейнер при їх вичерпанні.

У даному розділі були розглянуті принципи організації та властивості еластичних систем. Зокрема, розглянуті концепція еластичної системи та три основних способи реалізації еластичних систем у хмарному середовищі: у центрах обробки даних (сервери), на хмарних платформах

(віртуальні машини) та на рівні оркестрації (контейнери).

Висвітлені методи управління обчислювальними ресурсами еластичних систем, орієнтовані на:

- захист системи від перевантажень;
- зниження вартості володіння;
- аналіз характеру навантажувального процесу;
- оцінку необхідної кількості ресурсів;
- планування розподілу обчислювальних ресурсів.

Зроблений висновок щодо необхідності короткочасного прогнозування еластичного трафіка в мережах промислового Інтернету речей.

### 3 КОРОТКОСТРОКОВЕ КВАНТОВЕ ПРОГНОЗУВАННЯ ЕЛАСТИЧНОГО ТРАФІКУ В МЕРЕЖАХ ПРОМИСЛОВОГО ІНТЕРНЕТУ РЕЧЕЙ

#### 3.1 Методи аналізу трафіку промислового Інтернету речей

Велика кількість паралельних, різнорідних за характером, інформаційних потоків, що надходять з кінцевих пристроїв (датчиків, вимірювальних пристроїв, відеокамер та інших) промислового Інтернету речей (ІоТ) призводять до виникнення складної динаміки процесів передачі даних. Зміни величин затримок у каналах передачі даних, виникаючі при цьому біфуркаційні явища, призводять до появи нестационарності в процесах пакетної комутації при передачі даних для аналізу, обробки та зберігання у хмару. Це безпосередньо впливає на характер зміни величин затримок у каналах передачі даних, що призводить до суттєвих відхилень від середніх значень (рисунок 3.1).

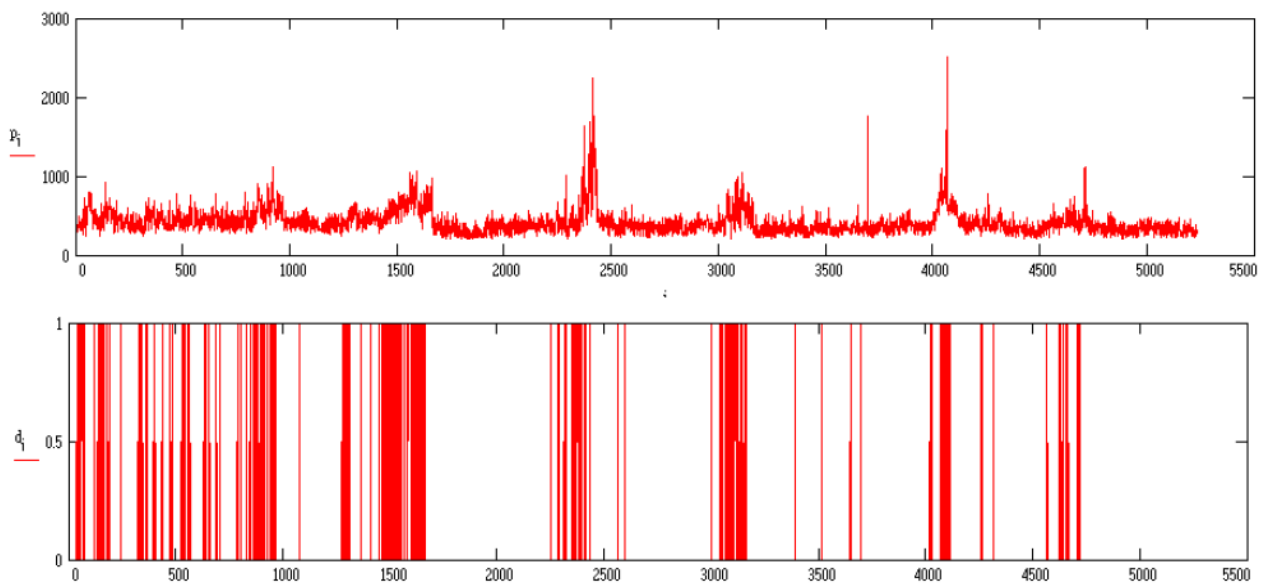


Рисунок 3.1 – Характер зміни затримок під час передачі даних ІоТ  
до хмарного середовища

Для дослідження такого агрегованого трафіка з наявністю флуктуацій широко застосовуються підходи, що ґрунтуються на аналізі виникаючих фрактальних процесів та використанні концепції активного мережного середовища. В результаті мережа промислового Інтернету речей як об'єкт управління розглядається під впливом численних регулюючих та обурювальних стохастичних впливів, що формують локальні флуктуації інтенсивності трафіку, створюваного кінцевими пристроями.

У зв'язку з цим особливої актуальності набувають питання аналізу наслідків впливу таких флуктуацій на зміну основних часових характеристик даного процесу. При цьому ключовою ланкою в структурі розподіленого управління процесом є система прогнозування трафіку ПоТ, що обробляється еластичними системами (еластичного трафіку), в якій враховуються особливості стохастичної та динамічної складових трафіку.

Одна з характерних властивостей досліджуваного процесу – масштабна інваріантність статистичних характеристик, пов'язана з проявом фрактальної самоподібності, що виникає внаслідок наявності довготривалих статистичних залежностей (ДТСЗ) в агрегованому інформаційному потоці. Ступінь впливу ДВЗС на проходження еластичного трафіку визначається тим, як успішно планувальник еластичної системи може впливати на зміну ресурсних складових залежно від прогнозованих змін стану агрегованого трафіку. Помилки в прогнозуванні можуть призвести до виникнення локальних перевантажень та невиправданих заримок в обробці інформації ПоТ.

На рисунку 3.2 показано логарифмічну функцію залежності ймовірності дотримання часових вимог системи ПоТ ( $P$ ) від умовного розміру затримки обробки наданої інформації ( $Tz$ ). Спостережуваний характер зміни отриманої залежності узгоджується з моделями ДТСЗ, тобто. ДТСЗ впливає на характер міжмережної взаємодії, що необхідно враховувати під час формування технічного та програмного забезпечення системи промислового Інтернету речей. Одна з основних проблем, що виникають у процесі управління еластичним трафіком, – своєчасне отримання інформації

про необхідність додаткового горизонтального масштабування. Оскільки протоколи маршрутизації використовують тільки локальну інформацію про проходження агрегованого трафіку, то індикатором необхідності завчасного масштабування у обслуговуючій хмарі може бути прогноз щодо можливого перевантаження.

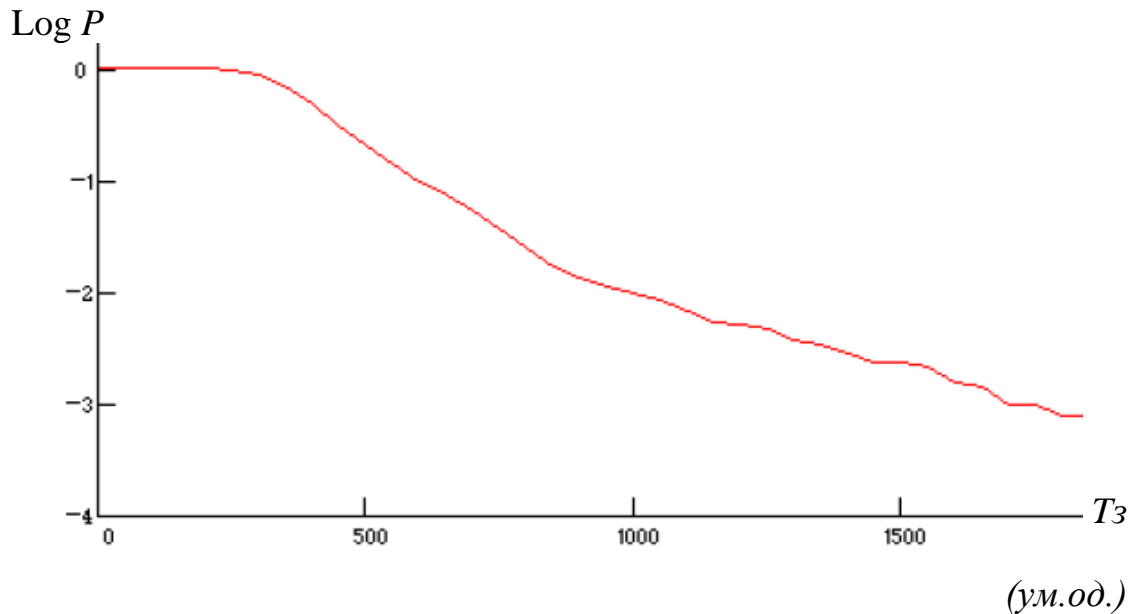


Рисунок 3.2 – Аналіз ймовірності дотримання часових вимог системи ПоТ

Вплив флуктуацій агрегованого трафіку на мережну продуктивність може бути суттєво знижено за допомогою введення ефективних методів прогнозування характеру змін трафіка.

У низці досліджень, які виявили наявність ДТСЗ в агрегованому трафіку ПоТ, були запропоновані різні моделі, засновані на узагальненні властивостей класичного броунівського руху  $B(t)$ , що дозволяє об'єднати різні аспекти динамічної та стохастичної поведінки складних об'єктів на основі простих і наочних математичних моделей. Зазвичай процес  $B(t)$  сприймається як функція однієї змінної  $t$ , приріст якої має гаусівський розподіл із нульовим математичним сподіванням і асимптотичною дисперсією, тобто

$$M[\Delta B_H^2(t)] \sim \Delta t^{2H}, \quad (3.1)$$

де  $\Delta t$  – часовий інтервал спостереження;  $H$  – параметр Нерста,  $0 < H < 1$ .

Експериментальним підтвердженням статистично протяжного характеру мережних процесів IoT може бути графік (Рисунок 3.3), що складається з 200 агрегованих у масштабі 25:1 відліків усередненого потоку даних з кінцевих пристроїв промислового Інтернету речей, знятого на еластичному сегменті хмарного середовища.

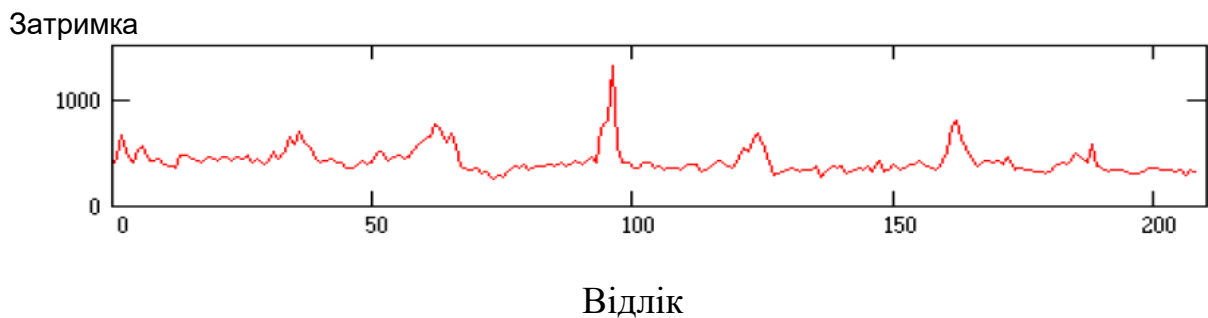


Рисунок 3.3 – Агрегований трафік у масштабі 25:1, побудований за 5000 відліками часу підтвердження приходу пакетів

В агрегованому трафіку чітко виділяється випадкова складова дуже великих відхилень, кореляційна пов'язаність яких зберігається на тривалому інтервалі спостереження. У цих умовах використання класичної формули Ерланга, в якій враховується лише спільний вплив параметрів завантаженості (інтенсивність викликів  $\mu$  та середній час заняття лінії  $t$ ), призводить до суттєво неточних оцінок. Похибка таких розрахунків пов'язані з тим, що в умовах інтегрованого навантаження не проявляється ефект "згладжування", а трафік, що спостерігається, характеризується високим ступенем нерівномірності (берстності) в широкому діапазоні інтервалів агрегування.

З позицій теорії управління мережа IoT є складним нестационарним динамічним об'єктом, або "чорною скринькою", характеристики якого змінюються в часі випадковим чином. У цьому контексті термін "чорний ящик" має кілька аспектів:

- а) структурний (вузли та зв'язки між ними);

- б) функціональний (методи та форми передачі даних);
- в) технологічний (ресурси та засоби передачі);
- г) сервісний (джерела та споживачі інформації, що передається),

що характеризують передачу інформації. Це дозволяє розглядати мережу IoT як множину взаємодіючих компонентів, які у свою чергу складаються з послідовності вкладених підрівнів. Один з можливих методів опису процесів у таких системах пов'язаний з використанням апарату дробового інтегрування та засобів фрактального аналізу. З формальної точки зору йдеться про синтез спеціального класу лінійних фільтрів, які використовують операцію дробового інтегрування для обчислення операції згортки вхідних та вихідних сигналів динамічних систем. Також для здійснення прогнозу використовують вейвлет-перетворення часових ділянок трафіку.

Але всі наведені вище підходи до короткострокового прогнозування не задовольняють вимоги до прогнозу еластичного трафіку в мережах промислового Інтернету речей: часовий інтервал, на якому формується необхідний масив відліків, порівняно невеликий, тому маємо невелику кількість відліків та суттєво обмежений час для формування прогнозу. Цих недоліків позбавлений квантовий метод, котрий розглянутий у наступному підрозділі.

### 3.2 Квантовий підхід до прогнозу еластичного трафіку

Розглянемо швидкий спосіб прогнозу на малих інтервалах часу, який використовує особливості еластичного агрегованого трафіку IoT. При великій кількості складових агрегації такий трафік набуває властивості самоподібності. Теоретичні та експериментальні дослідження самоподібних процесів вказують на можливість передбачення їхнього стану в майбутньому що зумовлено кореляційною структурою, яка присутня у довгостроково залежних процесах. Ця властивість може бути використана і для

передбачення майбутньої поведінки трафіку на невеликих масштабах часу, які адекватні процесам, що відбуваються в мережі під час обміну інформацією, і дозволяють застосовувати прогноз майбутньої поведінки трафіку в межах цього часу.

Агрегований трафік PoT завжди можна представити як суму 2 складових:

$$F_{PoT} = F_{min} + \Delta F, \quad (3.2)$$

де  $F_{min}$  – постійний гарантований обсяг трафіку, що надходить з кінцевих пристроїв PoT;  $\Delta F$  – змінна складова обсягу трафіку, яка впливає на параметри горизонтального масштабування, яка і підлягає короткостроковому прогнозуванню та змінюється в границях від 0 до  $\Delta F_{max}$  умовних одиниць обсягу трафіку.

Нехай базовий елемент еластичної хмарної системи (квант) забезпечує обробку  $f$  умовних одиниць обсягу трафіку. Тоді максимальна кількість додаткових квантів розраховується як

$$N = \left\lceil \frac{\Delta F_{max}}{f} \right\rceil, \quad (3.3)$$

де  $\lceil \cdot \rceil$  – операція отримання найменшого цілого числа, не меншого, ніж задане.

Прогнозування здійснюється наступним чином. Інтервал часу оцінки сегментується на рівні підінтервали тривалістю  $\tau$ :

$$\tau = n \cdot \Delta t, \quad (3.4)$$

де  $\Delta t$  – довжина фіксованого підінтервалу часу між двома сусідніми відліками,  $n$  – кількість відліків на одному підінтервалі, що обирається таким чином, щоб обов'язково виконувалась нерівність

$$\tau \geq \tau_0, \quad (3.5)$$

де  $\tau_0$  – час, необхідний для проведення горизонтального масштабування.

Для кожного такого  $j$ -го підінтервалу серед отриманих відліків обирається максимальне значення інтенсивності трафіку  $\lambda_j$  та обирається відповідний квант  $i \in \overline{1, N}$  за умови

$$f \cdot (i-1) < \lambda_j \leq f \cdot i. \quad (3.6)$$

Тоді кожний  $j$ -і підінтервал однозначно асоціюється з квантом  $i_j \in \overline{1, N}$ . Це дозволяє для кожної пари сусідніх підінтервалів  $j$  та  $j+1$  визначити міжквантовий перехід  $i_j \rightarrow i_{j+1}$ .

Приклад визначення міжквантових переходів для 10 квантів графічно наведений на Рисунок 3.4.

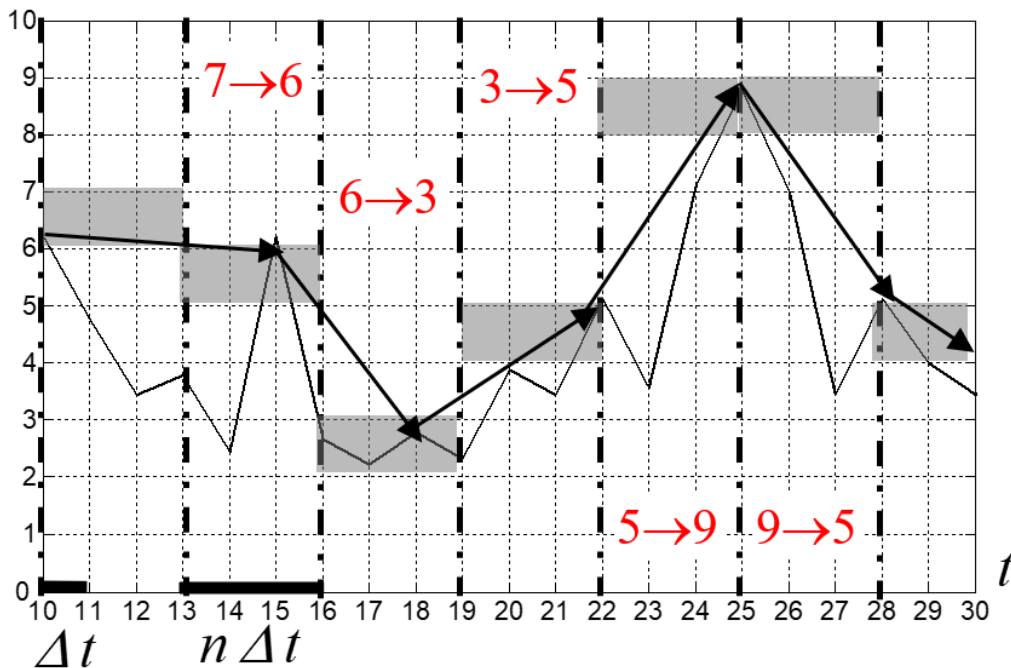


Рисунок 3.4 – Процес квантування трафіку ПоТ

Сформуємо тепер матрицю частот міжквантових переходів  $W = (w_{\xi,\eta})$ , де  $w_{\xi,\eta}$  – кількість переходів з  $\xi$ -го рівня квантування на  $\eta$ -й за час, що розглядається. Далі пронормуємо матрицю  $W$  по рядках

$$W_{\eta}^{(n)} = \left( w_{\xi,\eta}^{(n)} = w_{\xi,\eta} / \sum_{\xi=1}^{\ell} w_{\xi,\eta} \right). \quad (3.7)$$

При цьому  $\xi$  рядок матриці  $W_{\eta}^{(n)}$  можна розглядати як емпіричну дискретну функцію щільності розподілу ймовірностей переходів з сегмента рівня  $\xi$ .

Тобто можна розглядати щільність розподілу ймовірностей переходів з одного рівня квантування на інший, відповідно до якої прогноуються значення для горизонтального масштабування, достатні для обробки еластичного агрегованого трафіку ПоТ, а кожний ненульовий рядок матриці  $W_{\eta}^{(n)}$  можна відобразити дискретним полігоном значень, наприклад, як на рисунку 3.5 (а – д).

На об'єднаному графіку (Рисунок 3.5, е) бачимо зсув максимальних значень ймовірності переходу. Такий зсув є характерним саме для самоподібних процесів і характеризує можливість прогнозування.

Отже, для виконання процедури горизонтального масштабування на основі короткострокового прогнозу можна сформувати вектор міжквантових переходів.

$$V = \left( v_{\xi} \mid \forall (\eta \in \overline{1, N}) \Rightarrow w_{\xi,\eta} \leq w_{v_{\xi},\eta} \right), \quad (3.4)$$

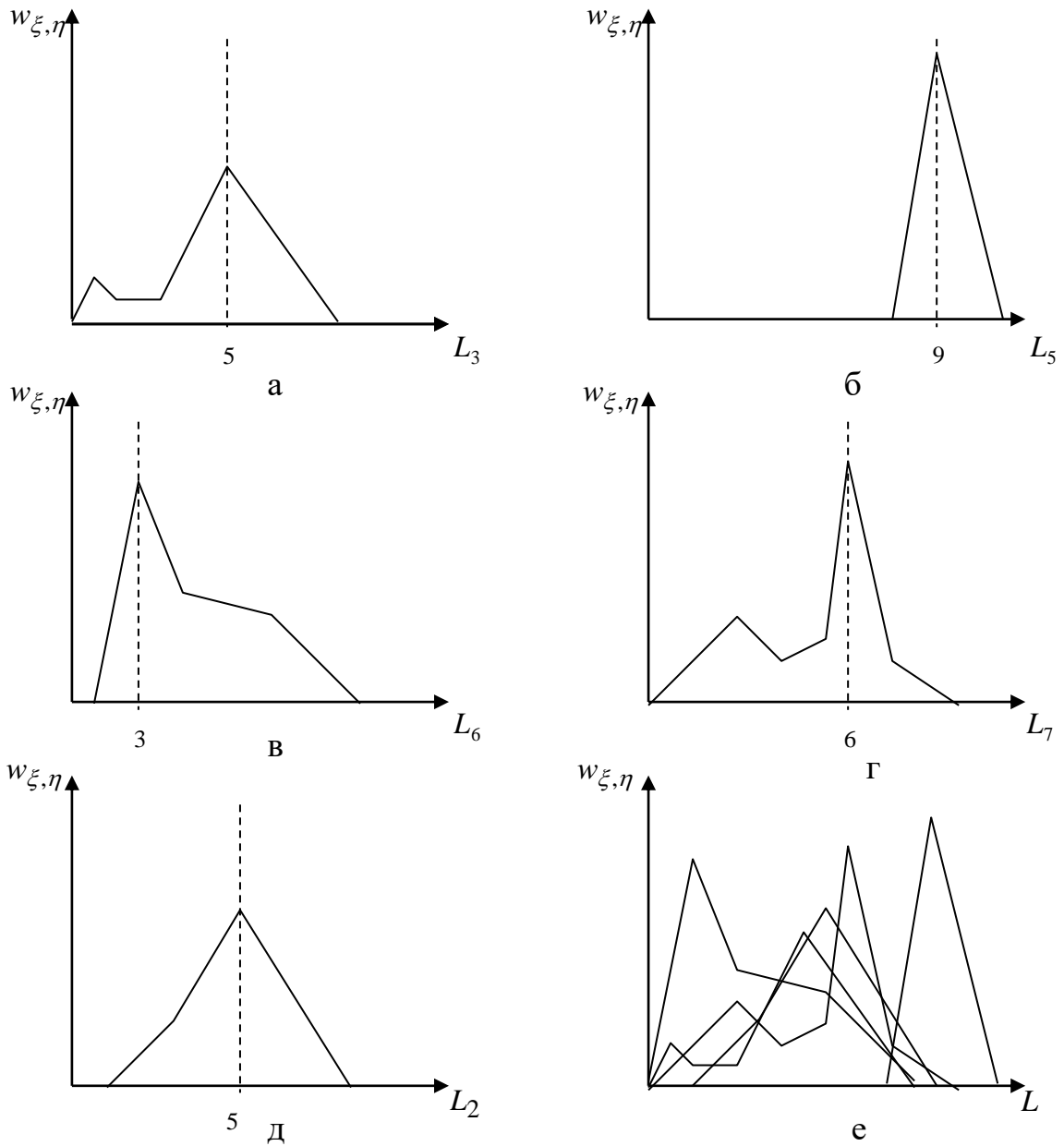


Рисунок 3.5 – Дискретні полігони розподілу ймовірностей міжквантових переходів: а – з рівня 3 до рівня 5; б – з рівня 5 до рівня 9; в – з рівня 6 до рівня 3; г – з рівня 7 до рівня 6; д – з рівня 9 до рівня 5; е – об'єднаний графік полігонів

Алгоритм горизонтального масштабування на основі короткострокового квантового прогнозування еластичного трафіку промислового інтернету речей розглянемо далі.

1 Нехай остання, що надійшла до хмари компонента трафіку ПоТ характеризується квантом  $i^*$ . Розглядаємо елемент поточного вектору  $V(i^*)$ .

2 Якщо  $V(i^*) = 0$  або  $V(i^*) = i^*$ , то горизонтальне масштабування не відбувається.

3 Якщо  $V(i^*) > i^*$ , то відразу запускається процедура горизонтального масштабування з розширенням на  $d^+ = V(i^*) - i^*$  базових елементів еластичної хмарної системи.

4 Якщо  $V(i^*) < i^*$ , то планувальник робить відмітку, що даний процес використовує  $d^- = i^* - V(i^*)$  надлишкових базових елементів еластичної хмарної системи, які можна виводити з даного процесу.

Відмітимо, що незалежно від того, здійснився прогноз чи ні, після надходження чергового елемента здійснюється корекція матриці і вектора міжквантових переходів.

### 3.3 Приклад прогнозування ділянки агрегованого трафіка ПоТ

Розглянемо ділянку агрегованого трафіка ПоТ (рисунок 3.6).

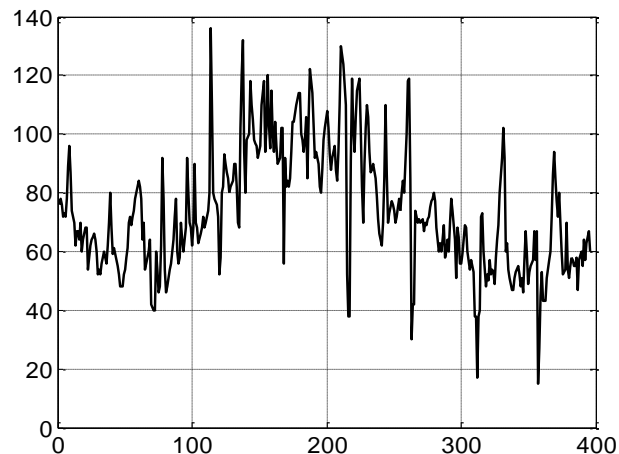


Рисунок 3.6 – Ділянка агрегованого трафіка ПоТ

Проведемо прогнозування міжквантових переходів на цій ділянці. На рисунку 3.7 представлені дискретні полігони значень розподілу міжквантових переходів.

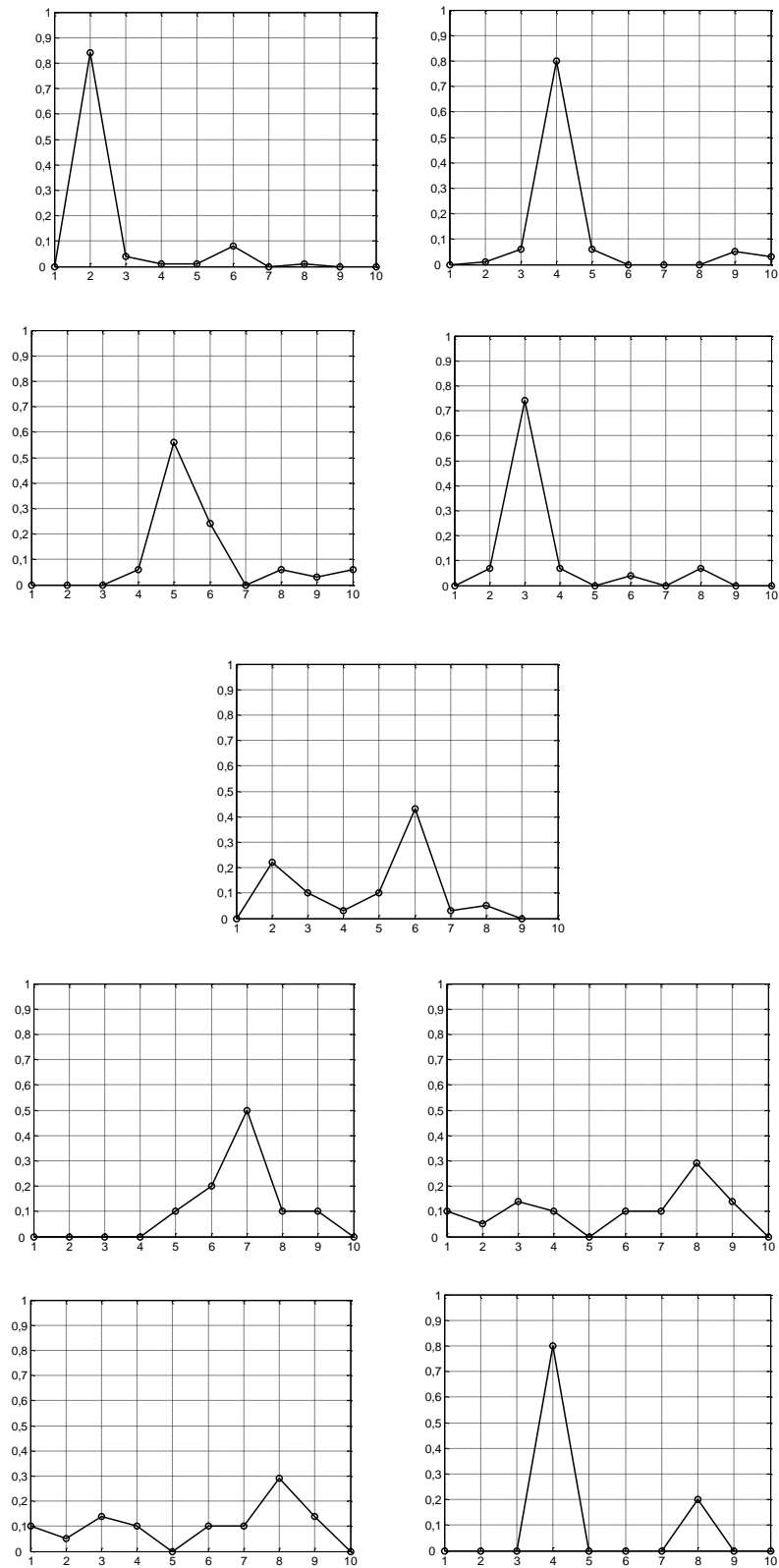


Рисунок 3.7 – Дискретні полігони розподілу ймовірностей міжквантових переходів

Як видно з графіків рисунку 3.7, для даного процесу характерний зсув пікових значень, що підтверджує прогнозованість трафіку ПоТ. Особливо добре це можна побачити на графіках рисунка 3.8

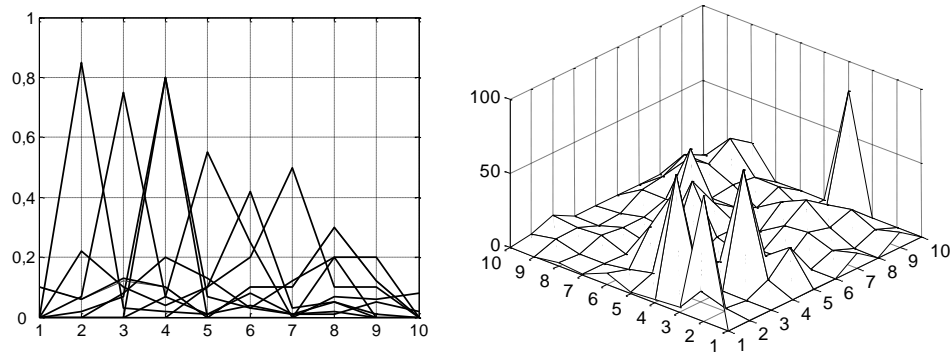


Рисунок 3.8 – Об'єднаний графік полігонів та тривимірна інтерпретація

В розділі запропонований метод короткострокового квантового прогнозування еластичного трафіку в мережах промислового Інтернету речей.

Проаналізовані існуючі методи аналізу трафіку промислового Інтернету речей. Доведено, що за обчислювальною складністю та оперативністю дані методи не задовольняють умовам для еластичного трафіка, тому що в даному випадку маємо невелику кількість відліків та суттєво обмежений час для формування прогнозу.

Запропонований квантовий підхід до прогнозу еластичного трафіку, який використовує специфічні властивості агрегованого та еластичного трафіка ПоТ і базується на формуванні найбільш ймовірного міжквантового переходу, що дозволяє завчасно запуснути відповідну процедуру горизонтального масштабування у хмарному середовищі.

Наведений приклад реалізації запропонованого підходу.

## ВИСНОВКИ

Сукупність отриманих у дипломній роботі результатів дозволило вирішити актуальне науково-технічне завдання, спрямоване на розробку підходу до прогнозування еластичного трафіка в мережах промислового Інтернету речей.

В результаті проведених досліджень отримані такі наукові та практичні результати:

1 Проведений аналіз можливостей та завдань сучасних систем Інтернету речей. Зокрема, відмічена роль Інтернету речей у сучасному світі, наведені темпи його зростання та розширення сфери застосування. Також наведена класифікація Інтернету речей, що поділяє його на дві великі категорії – споживчий Інтернет речей і промисловий (індустріальний) Інтернет речей. Обґрунтовані необхідність динамічного підходу до виділення еластичних хмарних обчислювальних ресурсів системам промислового Інтернету речей та необхідність в розробці методу короткочасного прогнозування трафіку промислового Інтернету речей, що набуває еластичної властивості

2 Визначені принципи організації та властивості еластичних систем. Зокрема, розглянуті концепція еластичної системи та тріє основних способи реалізації еластичних систем у хмарному середовищі: у центрах обробки даних (сервери), на хмарних платформах (віртуальні машини) та на рівні оркестрації (контейнери). Висвітлені методи управління обчислювальними ресурсами еластичних систем. Зроблений висновок щодо необхідності короткочасного прогнозування еластичного трафіка в мережах промислового Інтернету речей.

3 Розроблений підхід до короткострокового квантового прогнозування еластичного трафіку в мережах промислового Інтернету речей. Проаналізовані існуючі методи аналізу трафіку промислового Інтернету

речей. Доведено, що за обчислювальною складністю та оперативністю дані методи не задовольняють умовам для еластичного трафіка, тому що в даному випадку маємо невелику кількість відліків та суттєво обмежений час для формування прогнозу. Запропонований квантовий підхід до прогнозу еластичного трафіку, який використовує специфічні властивості агрегованого та еластичного трафіка IoT і базується на формуванні найбільш ймовірного міжквантового переходу, що дозволяє завчасно запуснути відповідну процедуру горизонтального масштабування у хмарному середовищі.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. А.О Сираєдоев, О.О. Можаяев. Дослідження споживчого і промислового інтернету речей.Тези доповідей 12-ої міжнародної науково-технічної конференції «Проблеми інформатизації», 21-22 листопада 2024р. Баку: ІСУ – Харків: ХНУРЕ, 2024, С.139.
2. Black U. Computer Networks: Protocols, Standards and Interfaces / U. Black. – Prentice-Hall, 2007. – 451 p.
3. Seng, K. P., Ang, L., & Ngharamike, E. (2022). Artificial intelligence Internet of Things: A new paradigm of distributed sensor networks. International Journal of Distributed Sensor Networks, 18(3), 155014772110628.
4. Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., & Zhao, W. (2017). A survey on Internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet of Things Journal, 4(5), 1125–1142.
5. Petrovska, I., Kuchuk, H. & Mozhaiev, M. (2022). Features of the distribution of computing resources in cloud systems. 2022 IEEE 4th KhPI Week on Advanced Technology, KhPI Week 2022 - Conference Proceedings, 03-07 October 2022, Code 183771.
6. Lorigo-Botran Tania, Miguel-Alonso Jose, Lozano Jose A. A review of auto-scaling techniques for elastic applications in cloud environments // Journal of grid computing. 2014. T. 12, № 4. С. 559–592.
7. Elastic application container: A lightweight approach for cloud resource provisioning / Sijin He, Li Guo, Yike Guo [и др.] // 2012 IEEE 26th International Conference on Advanced Information Networking and Applications / IEEE. 2012. С. 15–22.
8. Kuchuk, N., Kovalenko, A., Ruban, I., Shyshatskyi, A., Zakovorotnyi, O. & Sheviakov, I. (2023). Traffic Modeling for the Industrial Internet of NanoThings. 2023 IEEE 4th KhPI Week on Advanced Technology, KhPI Week 2023 - Conference Proceedings, 194480.

9. Naveen, S., & Kounte, M. R. (2019). Key Technologies and challenges in IoT Edge Computing. 2019 Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC).
10. Mouradian, C., Naboulsi, D., Yangui, S., Glitho, R., Morrow, M., & Polakos, P. (2018). A comprehensive survey on FoG Computing: State-of-the-Art and Research challenges. *IEEE Communications Surveys and Tutorials*, 20(1), 416–464.
11. Agarwal, S., Yadav, S., & Yadav, A. K. (2016b). An efficient architecture and algorithm for resource provisioning in Fog computing. *International Journal of Information Engineering and Electronic Business*, 8(1), 48–61.
12. Data center power cost optimization via workload modulation / Cheng Wang, Bhuvan Uргаonkar, Qian Wang [и др.] // 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing / IEEE. 2023. С. 260–263.
- Agarwal, S., Yadav, S., & Yadav, A. K. (2016b). An efficient architecture and algorithm for resource provisioning in Fog computing. *International Journal of Information Engineering and Electronic Business*, 8(1), 48–61.
13. Olifer V. Computer Networks: Principles, Technologies and Protocols for Network Design / V. Olifer, N. Olifer. – John Wiley & Sons, 2023. – 996 p.
14. Stallings W. Data and Computer Communications / W. Stallings. – Pearson Education, 2021. – 1023 p.
15. Kurose J. Computer Networking: A Top-Down Approach / J. Kurose, K. Ross. – 6th Edition. – Addison-Wesley, 2022. – 864 p.
16. Annabel Z. The Essential Guide to Telecommunications / Z. Annabel. – Prentice Hall, 2022. – 483 p.
17. Scherbo V. Local Area Networks Standards: Handbook / V. Scherbo, V. Kireichev, S. Samoylenko. – M.: Radio and Communications, 2020. – 316 p.
- Jennings F. Practical data communications: modems, networks and protocols / F. Jennings. – Blackwell Scientific, 2015. – 296 p.