

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)

Методи обробки великих даних в розподілених
хмарних системах

(тема)

Виконав:

студент II курсу, групи СПМ-22-6
Кипаренко Д.О.
(прізвище, ініціали)

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

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

Освітня програма Системне програмування
(повна назва освітньої програми)

Керівник: проф. Волк М.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2024 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-наукова
(освітньо-професійна або освітньо-наукова)
Освітня програма _____ Системне програмування
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту _____ Кипаренку Дмитру Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Методи обробки великих даних в розподілених хмарних системах

затверджена наказом по університету від “ 01 ” квітня 2024 р. № 257 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 15 червня 2024 р.

3. Вхідні дані до роботи _____

Моделі хмарних обчислень.

Постачальники хмарних послуг (AWS, GCP, Azure).

Методи оцінки продуктивності хмарних платформ.

4. Перелік питань, що потрібно опрацювати у роботі _____

Аналіз предметної області.

Методи організації обчислень у хмарному середовищі.

Застосування хмарних обчислень для управління енергоспоживанням при обробці великих даних.

Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Презентація 12 слайдів

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	02.04.24 – 04.04.24	
2	Розробка моделей	05.04.24 – 15.04.24	
3	Реалізація алгоритмів	16.04.24 – 28.04.24	
4	Розробка структури програмних засобів	29.04.24 – 15.05.24	
5	Розробка програмних модулів	16.05.24 – 25.05.24	
6	Оформлення матеріалів кваліфікаційної роботи	26.05.24 – 05.06.24	
7	Подання кваліфікаційної роботи керівникові та попередній захист	07.06.24 – 12.06.24	
8	Подання кваліфікаційної роботи на рецензування	13.06.24 – 15.06.24	

Дата видачі завдання 01 квітня 2024 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

проф. Волк М.О.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 66 с., 15 рис., 1 дод., 54 джерела.

МЕТОДИ, ХМАРНІ ОЧИСЛЕННЯ, ВЕЛИКІ ДАНІ, ПОТОКИ ЗАВДАНЬ, МОДЕЛІ ОБСЛУГОВУВАННЯ, ЕНЕРГОЕФЕКТИВНІСТЬ.

Традиційні паралельні обчислення для систем управління енергоспоживанням мають деякі проблеми, такі як час виконання, обчислювальна складність, затримки в моніторингу стану енергосистеми. Через ці обмеження управління даними стало критичним дослідженням і вузьким місцем. Щоб впоратися з цими обмеженнями, були запроваджені методології на основі хмарних обчислень для ефективного керування даними в системах управління живленням. У кваліфікаційній роботі розглядається концепція архітектури хмарних обчислень, яка може відповідати багаторівневим вимогам у режимі реального часу для покращення моніторингу та продуктивності, яка розроблена для різних сценаріїв застосування при моніторингу енергосистеми. Досліджуються рішення для хмарних обчислень на тлі великих даних, а нові моделі паралельного програмування Hadoop, Spark і Storm розглядаються в контексті ефективного використання. Ключові показники продуктивності програм хмарних обчислень, такі як вибірка даних, моделювання та аналіз конкурентоспроможності, були змодельовані шляхом застосування відповідних гіпотез. Пропонується нова концепція дизайну, розроблено рекомендації, зосереджені на інфраструктурі хмарних обчислень, а також методи керування великими даними в реальному часі в системі керування живленням, які вирішують проблеми інтелектуального аналізу даних.

ABSTRACT

Master's thesis: 66 pages, 15 figures, 1 appendice, 54 sources.

METHODS, CLOUD COMPUTING, BIG DATA, TASK FLOWS, SERVICE MODELS, ENERGY EFFICIENCY.

Traditional parallel computing for energy management systems has some problems, such as execution time, computational complexity, and delays in monitoring the state of the power system. Because of these limitations, data management has become a critical research and bottleneck. To cope with these limitations, cloud-based methodologies have been introduced to effectively manage data in power management systems. The thesis examines the concept of a cloud computing architecture that can meet multi-level requirements in real-time to improve monitoring and performance, which is designed for different application scenarios in power system monitoring. Big data cloud computing solutions are explored, and the new parallel programming models of Hadoop, Spark, and Storm are discussed in the context of efficient use. Key performance indicators of cloud computing applications, such as data sampling, modeling, and competitive analysis, were modeled by applying relevant hypotheses. A new design concept is proposed, guidelines are developed focusing on cloud computing infrastructure, and real-time big data management techniques in power management system that solve data mining challenges.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	11
1.1 Георозподілені хмарні центри обробки даних.....	11
1.2 Передумови дослідження	12
1.3 Моніторинг стану енергосистеми	16
1.4. Проблеми моніторингу енергосистем.....	19
1.5 Постановка мети та задач роботи.....	27
2 МЕТОДИ ОРГАНІЗАЦІЇ ОБЧИСЛЕНЬ У ХМАРНОМУ СЕРЕДОВИЩІ.....	29
2.1 Методи паралельних обчислень	29
2.2 Недоліки традиційних паралельних обчислень	31
2.3 Хмарні обчислення	35
2.4 Середовище хмарних обчислень	37
2.4.1 Технологія Hadoop	38
2.4.2 Технологія Spark	41
2.4.3 Технологія Storm	42
3 ЗАСТОСУВАННЯ ХМАРНИХ ОБЧИСЛЕНЬ ДЛЯ УПРАВЛІННЯ ЕНЕРГОСПОЖИВАННЯМ ПРИ ОБРОБЦІ ВЕЛИКИХ ДАНИХ.....	44
3.1 Порівняння паралельних обчислень із хмарними обчисленнями.....	44
3.2 Розподілені хмарні обчислення та паралельні обчислення.....	45
ВИСНОВКИ.....	53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	54
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	60

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ІТ – інформаційні технології

ПЗ – програмне забезпечення

ПК – персональний комп'ютер

ЦОД – центр обробки даних

ЦП – центральний процесор

AI – штучний інтелект (англ., Artificial intelligence)

API – інтерфейс програмування додатків (англ., Application Programming Interface)

AWS – Amazon Web Services

CPU – центральний процесор (англ., Central Processing Unit)

GCP – Google Cloud Platform

HPC – високопродуктивні обчислення (англ., High Performance Computing)

IaaS – інфраструктура як послуга (англ., Infrastructure as a Service)

IoT – інтернет речей (англ., Internet of Things)

ISA – промислова стандартна архітектура (англ., Industry Standard Architecture)

HTTP – протокол передачі гіпертексту (англ., HyperText Transfer Protocol)

MIMD – множинний потік команд, множинний потік даних (англ., Multiple Instruction, Multiple Data)

MIPS – Million Instructions Per Second

MISD – множинний потік команд, одиночний потік даних (англ., Multiple Instruction, Single Data)

MPI – інтерфейс передачі повідомлень (англ., Message Passing Interface)

MPICH – Message Passing Interface CHameleon

PaaS – платформа як послуга (англ., Platform as a service)

PE – Processing Element

SaaS – програмне забезпечення як послуга (англ., Software as a service)

SIMD – одиночний потік команд, множинний потік даних (англ., Single Instruction, Multiple Data)

SISD – одна команда, одне дане (англ., Single Instruction, Single Dat)

SOAP – протокол обміну повідомленнями (англ., Simple Object Access Protocol)

ВСТУП

На ранньому етапі розвитку технології моніторингу стану інформаційної системи, більшість систем моніторингу розроблялися для певного типу обладнання [1,2], і кожна система була розкидана та ізольована [3]. Це був інформаційний острів без обміну даними та взаємодії та не сприяв управлінню та комплексному аналізу даних моніторингу [4]. Крім того, апаратні ресурси, такі як мережа, обчислювальна техніка та засоби зберігання даних цих систем моніторингу було важко спільно використовувати, що призводило до марної витрати ІТ-ресурсів [5]. Таким чином, з'явилася інтегрована система управління, побудована в головній диспетчерській, яка може централізовано обробляти різноманітні дані моніторингу, зібрані різними пристроями моніторингу [6,7]. Щоб інтегрувати пристрої моніторингу різних специфікацій у централізовану систему моніторингу, необхідно розробити технічні регламенти і протоколи зв'язку, а також створити центри моніторингу для електропередачі та трансформаторного обладнання в місцевих компаніях [8].

Однак поточні обмеження пристроїв моніторингу дозволяють лише завантажувати оброблені спрощені дані в центр моніторингу, а частота збору даних не висока [9,10]. Із популяризацією та застосуванням високошвидкісних оптоволоконних мереж та технології бездротової передачі в електроенергетиці майбутній центр моніторингу стану енергетичного обладнання зможе отримувати панорамні дані моніторингу стану в режимі реального часу з більшої території та стати центром обробки даних для інтеграції даних і обміну інформацією [11]. Таким чином, обсяг зібраних даних у центрі моніторингу має перспективу у майбутньому, а можливості обробки інформації існуючої системи моніторингу будуть недостатніми для задоволення вимог щодо зберігання та обробки таких масивних даних [12,13].

Очевидно, що послідовний метод обробки вже давно не може задовольнити вимоги обробки великих обсягів даних [14,15]. Традиційна парадигма паралельних обчислень, заснована на високопродуктивних комп'ютерах, завжди була відповідальною за різноманітні обчислювальні проблеми, що виникають у наукових дослідженнях та інженерній практиці [16]. Також є багато недоліків у даних про стан електрообладнання [7]. В останні роки технологія хмарних обчислень виникла з паралельних обчислень і швидко розвивалася, а її численні переваги принесли нові ідеї щодо створення обчислювальних платформ центрів моніторингу, які привернули увагу вчених в енергетичній галузі [2,17]. Судячи зі статусу досліджень, більшість платформ хмарних обчислень, розроблених наразі для центрів моніторингу, базуються на єдиній структурі Nadoop, яка має певні обмеження. Nadoop добре справляється з пакетною обробкою великих даних, але він не може відповідати багатшим обчислювальним режимам моніторингу стану енергетичного обладнання, таким як потокове обчислення [18-19].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Георозподілені хмарні центри обробки даних

Обробка великих даних стикається з рядом проблем. Вона потребує значних ресурсних затрат, а саме – велику кількість процесорів, а відповідно і інфраструктури, яка їх обслуговує. Відомо, що сучасні датацентри потребляють значний обсяг електроенергії, яка витрачається на обчислення, охолодження, передачу даних, забезпечення стабільної, безвідмовної роботи, підтримку сховищ даних тощо.

Очевидно, що сучасні методи управління системами обробки великих даних повинні враховувати енергоспоживання. З іншого боку, сучасний енергоринок відрізняється наявністю різних альтернативних джерел живлення, які мають різну вартість, надійність, якість постачання, що може значно вплинути на вартість обчислень (рисунок 1.1).

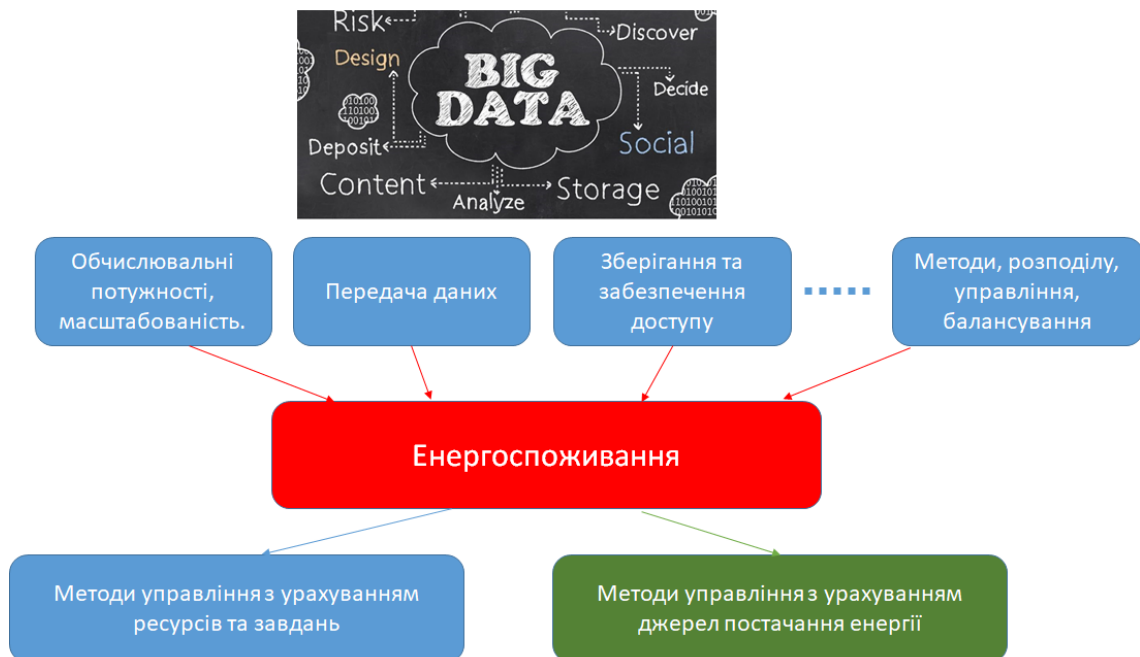


Рисунок 1.1 – Проблеми обробки великих даних

Насьогодні технології хмарних обчислень, що прийшли з галузі паралельних обчислень, швидко розвиваються, а її чисельні переваги приносять нові архітектури створення обчислювальних платформ центрів моніторингу, що можуть використовуватися у енергетичній галузі [2,17], як показано на рисунку 1.2. Судячи з сучасних досліджень, багато платформ хмарних обчислень, що розроблені наразі для енерго центрів, використовують структуру Hadoop, що має певні обмеження. Hadoop в основному використовує пакетну обробку великих даних, але він не має гнучкості з різними обчислювальними режимами моніторингу енергетичного обладнання, прикладом якого є потокове обчислення [18-19].

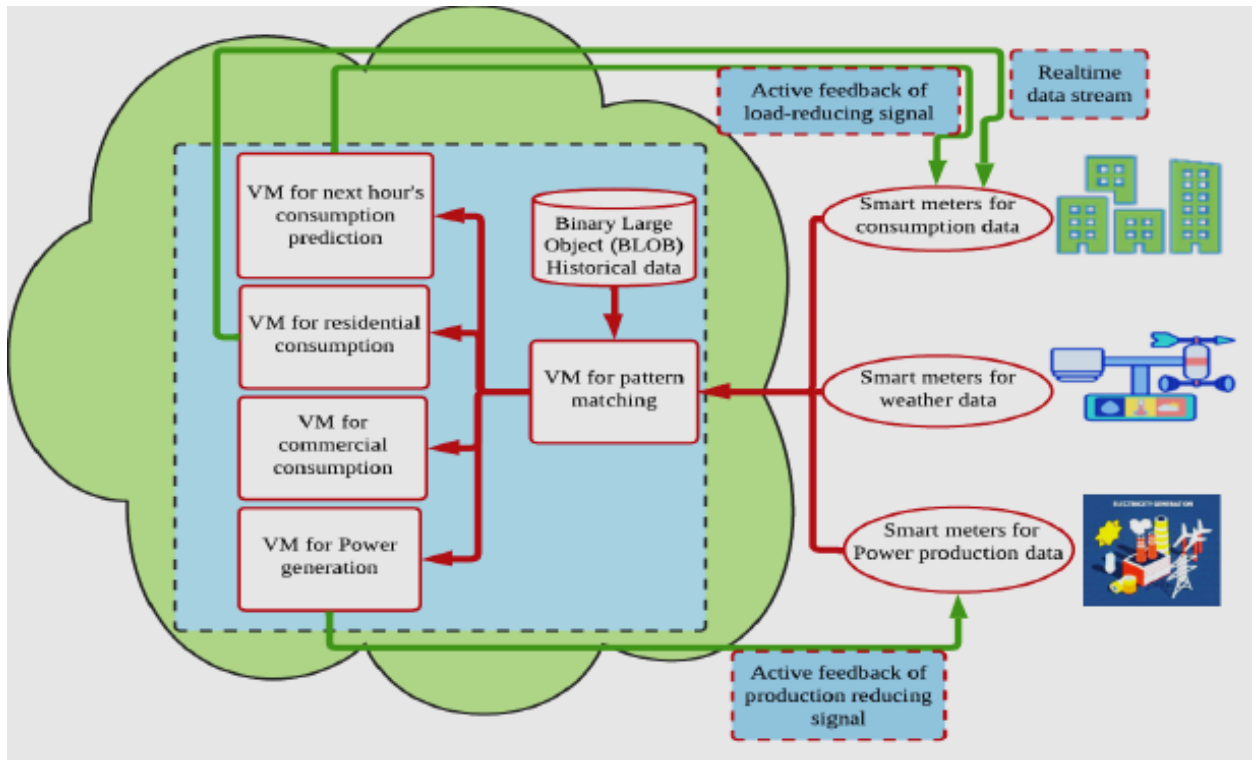


Рисунок 1.2 – Обробка даних в енергосистемі

1.2 Передумови дослідження

Диспетчерський центр електромережі розвивається в напрямку просування технології онлайн-моніторингу, яка об'єднує регулювання та

контроль електроенергії. Дані про робочий стан енергетичного обладнання надалі надсилатимуться до центру регулювання. Існуючі системи диспетчерського контролю та збору даних із системами керування електроенергією (SCADA/EMS) є складними для роботи з величезною кількістю даних моніторингу [20]. Для використання пристроїв моніторингу різних виробників для підключення до єдиної централізованої системи моніторингу, у багатьох мережевих компаніях створено центри моніторингу обладнання для передачі та трансформації електроенергії [2]. Незважаючи на те, що технологія веб-сервісу зменшує складність інтеграції даних, проблеми залишаються у задоволенні вимог електроенергетичних компаній до даних у реальному часі [14,15].

Таким чином, поточна система моніторингу отримує кваліфіковані дані, оброблені локально в пристрої моніторингу. Наприклад, перед завантаженням пристрій моніторингу повинен обробляти сигнал часткового розряду високовольтного електричного обладнання на кількість розрядів, піковий об'єм розряду та відповідну фазу розряду [21–23]. Завантаження «знайомих даних» замість «необроблених даних» може зменшити витрати на передачу по мережі та витрати на зберігання центру моніторингу [24]. Однак центр моніторингу, який об'єднує дані з різних специфікацій пристроїв моніторингу, все ще стикається зі складною проблемою поглибленої оцінки стану та збою діагностики цільового пристрою [25,26].

Великі дані моніторингу заповняють віддалений центр моніторингу енергетичного обладнання, що призведе до важких завдань зберігання, обробки та аналізу. Дані моніторингу стану енергетичного обладнання показали основні характеристики великих даних: масивні заобсягом, різноманітні за типами, швидкі зміни та низька щільність значень. Завдання обробки та аналізу даних моніторингу будуть передані з розподілених пристроїв моніторингу до централізованих центрів моніторингу, що спрощує конфігурацію апаратного та програмного забезпечення пристроїв моніторингу та сприяє гнучкому розширенню інфраструктури центру

моніторингу [27]. Компанія, яка здійснює моніторинг роботи віддалених приладів, перейшла від попередньої обробки даних і завантаження спрощених наборів даних до використання сіток даних пам'яті та інших технологій для отримання та зберігання необроблених даних датчиків [9]. Широке застосування високошвидкісного оптоволоконного зв'язку в енергетиці забезпечує рішення для передачі масивних даних [11].

Традиційна технологія обробки даних зіткнулася з вузькими місцями під час нинішнього вибухового зростання даних. Вона не може задовольнити потреби енергетики в аналізі для швидкого отримання знань та інформації з масивних даних. Інформація про енергетичну галузь – це дослідження та застосування енергетичних технологій великих даних. Це неминуча вимога для розвитку технологій та інтелекту. Інтелектуальний аналіз даних може комплексно використовувати відповідні алгоритми для обробки великої кількості даних і виявлення прихованої цінної інформації, яка може реалізувати швидке перетворення даних у знання, а потім значення [28–32].

Однак, поточний обсяг даних швидко зростає, і традиційні алгоритми, засновані на одновузловому послідовному видобутку даних, більше не підходять для масових потреб у даних. Як платформа розподіленої обробки даних, хмарні обчислення можуть інтегрувати багато комп'ютерних ресурсів і значно розширити технічні можливості. Він більше підходить для обробки великих даних, ніж звичайні алгоритми [14]. Крім того, моделі хмарних обчислень і платформи хмарних обчислень не мають високих вимог до мережеских вузлів, а звичайні комп'ютери також можуть брати участь у хмарних обчисленнях, що певною мірою знижує складність і вартість побудови хмарної платформи [33–35].

Останніми роками хмарним обчисленням приділяється більше уваги завдяки обчислювальній моделі, яка є інтегрованою з розробки звичайних комп'ютерів і технологічних мереж, таких як розподілені обчислення, паралельні обчислення, мережеве зберігання тощо [2,19,36].

Технології віртуалізації, розподіленого зберігання та паралельних обчислень у хмарних обчисленнях дають нові ідеї для побудови обчислювальних платформ для моніторингу стану енергетичного обладнання ЦОД. Можна інтегрувати існуючі базові обчислювальні засоби підприємств електроенергетики та забезпечити надійну, стабільну та потужну підтримку обчислювальної ємності та зберігання, що є корисним для моніторингу енергетичного обладнання онлайн у широкому діапазоні та географічної віддаленості [37,38].

Моніторинг і збір інформації покращують аналіз у реальному часі та можливості інтелектуальної діагностики [19,39]. Зі швидким розвитком технологій хмарних обчислень з'явилися різноманітні високонадійні та масштабовані системи обробки великих даних, такі як Hadoop, Spark і Storm, які забезпечують сприятливі інструменти для централізованої обробки великомасштабних даних моніторингу енергетичного обладнання [40–42].

Хоча ці нові моделі пропонують уніфікований інтерфейс програмування та захищають більше деталей низького рівня ніж традиційні моделі програмування паралельних обчислень, виникають питання їх впровадження в обробку даних центру моніторингу енергетичного обладнання.

Поєднання певного професійного досвіду для вирішення практичних завдань і різних застосувань високого рівня в системі моніторингу все ще є темою, яка заслуговує на вивчення [7,16]. Характеристики масивних даних в інтелектуальній мережі. Багато досліджень узгоджуються з широко поширеними даними, та сконцентровані на п'яти основних напрямках, як показано на рисунку 1.3 [43–46].

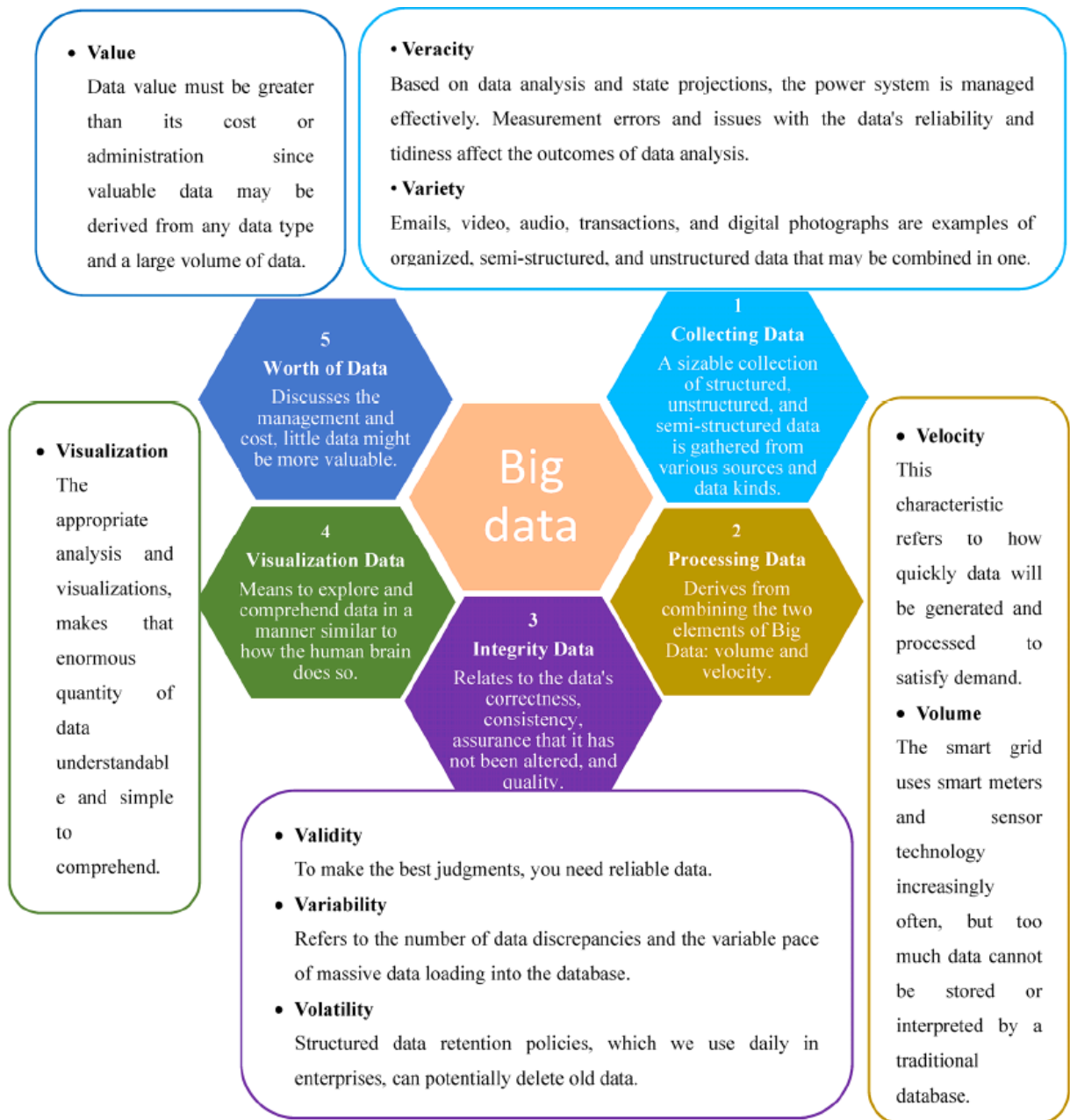


Рисунок 1.3 – П'ять категорій великих даних із їхніми характеристиками

1.3 Моніторинг стану енергосистеми

Моніторинг в режимі онлайн – це безперервний або періодичний автоматичний збір електричної, фізичної, хімічної та іншої інформації про стан об'єкта моніторингу за допомогою пристроїв моніторингу, встановлених на об'єкті моніторингу або поблизу нього без відключення живлення. Він може передати польову шину, Ethernet, бездротовий зв'язок та

інші методи зв'язку, передавати дані про стан у систему віддаленого моніторингу для централізованого зберігання та обробки [18,47], як показано на рисунку 1.4. Система онлайн-моніторингу використовує розширену обробку інформації і технологію діагностики для обробки та комплексного аналізу даних про стан, які можуть передбачити залишковий термін служби цільового обладнання та забезпечити базу даних і основу для підтримки стану [48].

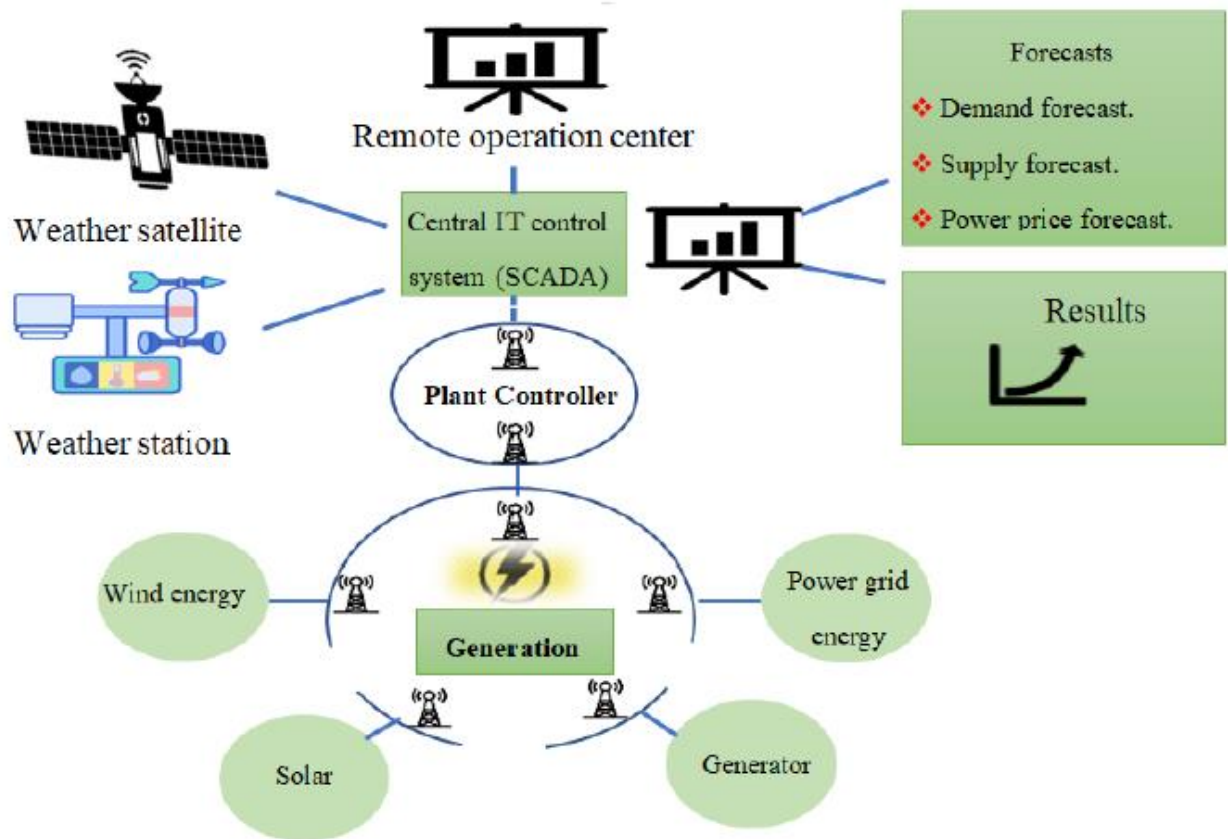


Рисунок 1.4 – Моніторинг енергосистеми

У той же час з'явилися випробувальні прилади, здатні проводити інфрачервоні, ультразвукові та інші неелектричні вимірювання. З 1990-х років сенсорна технологія, комп'ютерна технологія та технологія отримання цифрових сигналів вивели онлайн-моніторинг на новий етап. З'явилася багатофункціональна мікрокомп'ютерна система онлайн-моніторингу, яка реалізувала моніторинг більшої кількості параметрів обладнання, таких як

коефіцієнт діелектричних втрат, електрична ємність, струм витоку, частковий розряд тощо. Ця система моніторингу має більш повні функції та покращену конфігурацію програмного та апаратного забезпечення, який може реалізувати обробку даних, аналіз, діагностику, сигналізацію та візуалізацію результатів [49]. Інтеграція, автоматизація та інтелект стали поточним напрямком розвитку технологій онлайн-моніторингу [50]. Однак більшість існуючих систем моніторингу електроенергії перебувають у стадії ізольованої роботи, а використання великих обсягів отриманих даних є низьким [4].

В останні роки він може лише дозволити підсистемам часткового моніторингу завантажувати спрощені «знайомі дані» і не може передавати вибірккові дані високочастотних сигналів. Такі, як сигнал розряду електричного обладнання, і ці дані корисні для діагностики несправностей такого обладнання, як трансформатори та GIS-перемикачі [49]. Популяризація та застосування технології високошвидкісного оптоволоконного зв'язку дозволить передавати великі обсяги даних [50,51].

Майбутня система моніторингу енергетичного обладнання отримуватиме панорамні дані моніторингу стану в реальному часі від широкого діапазону енергетичного обладнання та стане даними, які можуть реалізувати інтеграцію даних та центр обміну інформацією [52].

Більшість традиційних мережевих систем SCADA збирають дані за допомогою циклічного запиту, який не може збирати дані постійного моніторингу енергетичного обладнання. Задовольнити вимоги до обробки в режимі реального часу та динамічного доступу до поточних великих даних підвищує складність моніторингу обладнання [20]. Нещодавня поява технології хмарних обчислень може вселити надію на подолання труднощів зі збору, зберігання і обробкою великих даних для моніторингу енергетичного обладнання [51]. У роботі [19] запропонована архітектура управління інформацією та обчислювальної платформи на основі технології хмарних обчислень для моніторингу стану інтелектуальної мережі, у якій

обчислювальний рівень використовує лише структуру Hadoop. в [53] обговорили доцільність і необхідність використання енергетичної системи за допомогою хмарних обчислень, пояснили особливості та цілі структури енергетичної системи, що використовує хмарний обчислювальний центр, щоб запропонувати структуру системи на основі імітаційних обчислень як прикладу для задоволення потреб майбутньої розумної сітки для зберігання, спільного використання та обробки великої панорамної інформації. У роботі [51] запропонована обчислювальна платформа, що забезпечує сховище та обчислювальне середовище для бізнес-додатків у центрі обробки даних. У роботі [18] об'єднали характеристики інтелектуальних даних моніторингу стану підстанції, вивчили платформу обробки даних онлайн-моніторингу обладнання підстанції на базі Hadoop і зосередилися на методах зберігання та швидких запитів моніторингу даних у розподіленій базі даних.

Автори роботи [19] досліджували інтелектуальну систему діагностики обладнання підстанції на основі хмарної платформи, надали різноманітні методи інтелектуальної діагностики та реалізували спільну діагностику несправностей та ієрархічну діагностику за допомогою стратегій злиття інформації.

1.4. Проблеми моніторингу енергосистем

Онлайн-моніторинг первинного інтелектуального енергетичного обладнання та традиційного енергетичного обладнання отримав різкий розвиток і став трендом [2,19]. Дані моніторингу стають все більш і більш масовими, і система онлайн-моніторингу енергетичного обладнання стикається з величезними технічними проблемами, такими як (у режимі реального часу, швидкі зміни, висока точність, різні програми (різні типи), аномальні дані, отримані від даних про енергопостачання, великомасштабні дані зі складною структурою, великими розмірами та масивними даними) [27].

Дослідницьку роботу можна грубо розділити на два види методів виявлення, заснованих на традиційних методах виявлення, і методів виявлення, заснованих на аналізі даних [14]. Традиційні методи виявлення ненормального силового навантаження зазвичай базуються на людському досвіді, оцінці стану, кривій навантаження, подібності та швидкості зміни навантаження [51]. Через низьку ефективність, сильну суб'єктивність і багато людських факторів цього методу. Цей метод підходить лише для аномальних даних із великими різкими змінами та не може виявити аномальні дані з неочевидними змінами [53]. З безперервним розвитком штучного інтелекту та теорії кластерного аналізу швидкість технології інтелектуального аналізу даних у виявленні даних про аномальне енергонавантаження також зростає [50]. Зазвичай можна грубо розділити на дві категорії: по-перше, метод виявлення аномальних значень із контрольованим навчанням, який вибирає частину даних про навантаження енергосистеми як навчальну вибірку, а потім використовує відповідний алгоритм, щоб зробити вибрані вибіркові дані та очікувані результати відповідає відповідним вимогам на основі опорних векторних машин, методів виявлення на основі штучних нейронних мереж і методів виявлення на основі дерев рішень [40]. По-друге, для методу виявлення ненормальних даних про потужне навантаження без контролю не потрібно вибирати частину історичних даних про потужне навантаження як навчальні зразки [41]. Цей тип методу виявлення аномальних даних зазвичай включає аналіз щільності, кластерний аналіз тощо.

Алгоритм виявлення викидів на основі відстані належить до неконтрольованого алгоритму. Цей метод легко зрозуміти та пояснити, і це один із найбільш репрезентативних методів у алгоритмах виявлення викидів на основі щільності, і він дає кращі результати при роботі з даними середньої та високої розмірності [51].

Метод виявлення даних про ненормальне навантаження на основі щільності все ще має проблему, оскільки деякі параметри визначаються суб'єктивними факторами, такими як людський досвід, що знижує точність

виявлення аномальних даних [42]. Метод виявлення аномальних значень на основі алгоритму кластеризації може більш точно розрізняти нормальні дані та аномальні дані в наборі даних електричного навантаження відповідно до характеристик даних електричного навантаження.

В даний час технологія хмарних обчислень досягла хороших результатів у сфері зберігання медичних даних, аналізу трафіку в режимі реального часу та аналізу погодних даних [37,53]. Вартість хмарних обчислень низька, і немає складних вимог до серверів створення кластеру [44]. Завдяки великому масштабу та швидкій швидкості обчислень хмарних обчислень, управління та аналіз даних моніторингу живлення може бути реалізовано більш ефективно завдяки інтеграції з традиційними методами інтелектуального аналізу даних [33,45].

Підводячи підсумок, можна сказати, що метод виявлення навантаження на потужність і аномальних даних, заснований на інтелектуальному аналізі даних і кластерному аналізі, став гарячою точкою досліджень за останні роки через його здатність глибше вивчати мінливий закон кривої навантаження та ефективно виявляти дані про ненормальні навантаження, а також певний результат досягнуто [12,36]. На практиці типи даних про потужне навантаження складні та різноманітні, і протиріччя між зростаючим масштабом даних про потужне навантаження та низькою ефективністю алгоритмів аналізу даних поступово стає помітним. Кожен алгоритм кластеризації має недоліки, такі як труднощі в оптимізації визначення початкових параметрів і висока чутливість [37]. Таким чином, відповідні алгоритми для паралельного виявлення даних про навантаження на потужність потребують подальшого вивчення. Усі наведені вище пояснення підсумовані на рисунку 1.5.

На даний момент дослідження застосування технології великих даних на основі хмарних обчислень в енергетиці все ще знаходяться на дослідницькій стадії [30]. Наразі більшість платформ хмарних обчислень, розроблених для центрів моніторингу, базуються на єдиній структурі Nadoop

і мають певні обмеження щодо зберігання даних до централізованої обробки, що призводить до надто тривалої затримки обробки [29], які не можуть відповідати вимогам даних онлайн-моніторингу [28,42]. Швидке та ефективно задоволення потоку даних стане тенденцією обробки інформації в майбутньому [12,18].

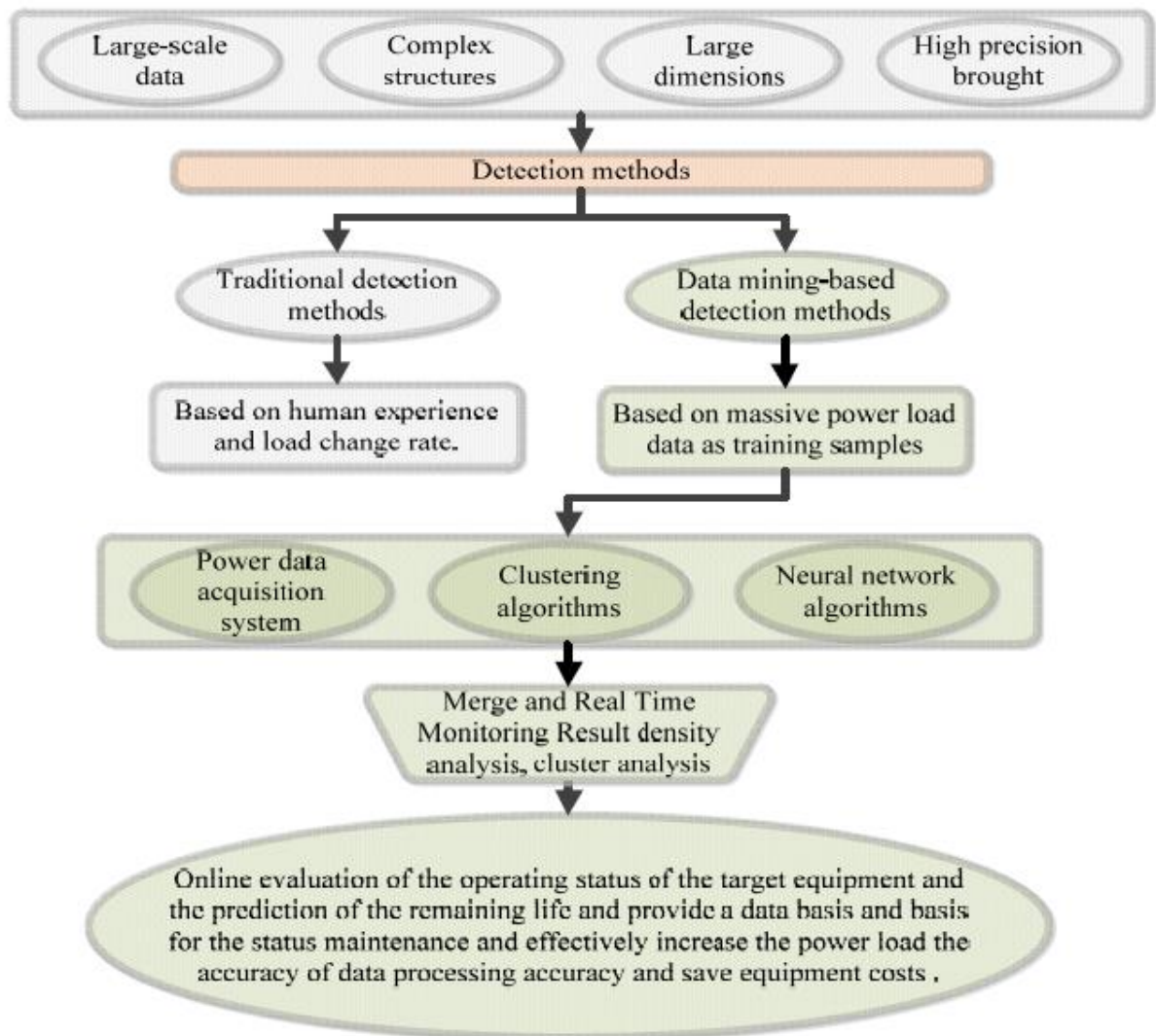


Рисунок 1.5 – Методи дослідження енергосистем

Незважаючи на кілька досліджень щодо швидкості, обсягу та різноманітності, повне та ефективно рішення наразі недоступне на ринку; найпопулярніший спосіб полягає у використанні системи керування базами даних (СУБД), яка може бути несумісною зі старими системами [49]. Швидка

обробка даних моніторингу та проблема аномального виявлення в режимі реального часу стають все більш актуальними [13].

Впровадження технології хмарних обчислень у сферу електроенергетики має важливе дослідницьке значення, але застосування в енергетиці все ще знаходиться на дослідницькій стадії, і необхідні більш глибокі дослідження, щоб застосувати їх у виробництві електроенергії [70]. Технологія хмарних обчислень привернула велику увагу завдяки високій продуктивності, але як використовувати технологію хмарних обчислень для великомасштабної обробки даних у реальному часі, ще не вивчено [11]. Крім того, багатоджерельний гетерогенний міський сенсорний доступ і технології управління даними забезпечують потужну підтримку для інтелектуального сприйняття та наукового управління в масштабі міста та можуть прискорити будівництво розумних міст або цифрових міст-побратимів з функціями віртуальної реальності [12–14].

Складність процесів часто зростала через недостатність окремих обчислювальних ресурсів через складну роботу комбінованої моделі прогнозування та зустрічі в режимі реального часу для інтелектуальних енергетичних систем [7,39,51]. Складність аналізу та вивчення масивних даних про споживання електроенергії призвела до необхідності розробки алгоритмів аналізу кластеризації в поєднанні з хмарними обчисленнями [49]. У більшості досліджень хмарних обчислень, використовується локальна система управління [46,47].

Тому не вистачає досліджень, які б використовували хмарні обчислення, щоб отримати вигоду від можливості ширшого спільного використання систем, які контролюють процес. Кілька досліджень запропонували продукти хмарних обчислень, такі як штучний інтелект (AI) [28] та Інтернет речей (IoT) [5] у своїх моделях, проте всі дослідження зосереджувалися лише на акумуляторах розумних транспортних засобів [47]. Тому це дослідження зосереджено на проблемі централізованої паралельної обробки та діагностики даних про стан енергосистеми на основі хмарних

обчислень і технології великих даних, оскільки важко задовольнити вимоги енергосистеми в реальному часі та створити безпечну, стабільну, економічно ефективну, зелену та екологічно чисту розумну мережу. Виклики керування живленням і моніторингу мають багато проблем; ця робота пропонує націлитися на три різні проблеми (рисунок 1.6).

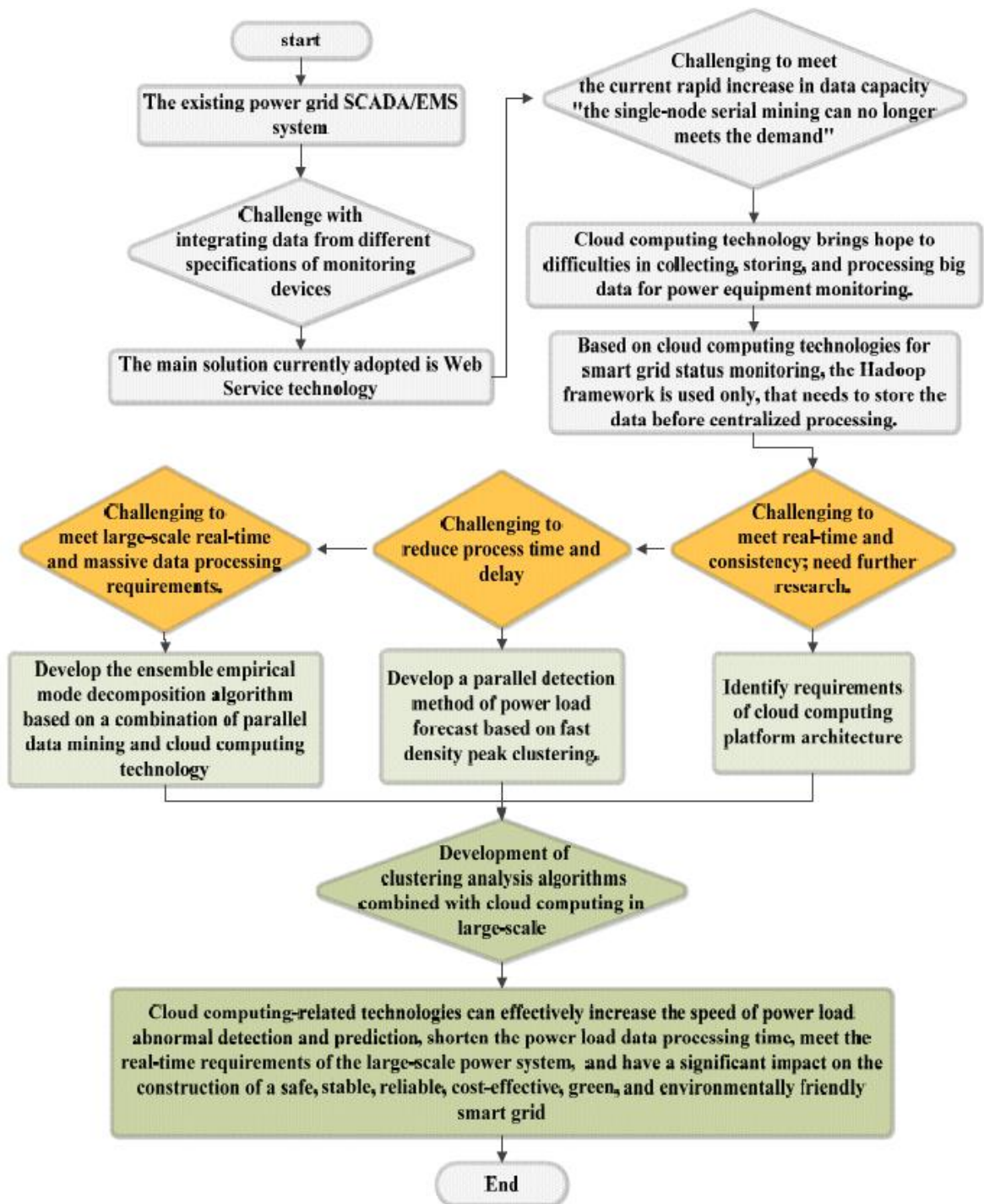


Рисунок 1.6 – Загальна блок-схема проблем і рішень запропонованої системи

По-перше, аналіз відповідності численним вимогам моніторингу стану енергосистеми в режимі реального часу. По-друге, аналіз слабкості традиційного інтелектуального аналізу даних на основі одновузлого послідовного інтелектуального аналізу. І, по-третє, недостатність існуючого алгоритму, який поєднує інтелектуальний аналіз даних і обчислювальну технологію для роботи з масивними даними.

Нарешті, хмарні обчислення можуть стати чудовим доповненням до інтелектуальної системи живлення, яка спрямована на вирішення проблеми масивних даних із великомасштабних областей розумної системи живлення. Внески до цієї оцінки літератури враховують вищезазначені питання. Щоб дати огляд аналітики великих даних за допомогою фреймворків хмарних обчислень для систем управління енергопостачанням на даний момент, ми пропонуємо першу універсальну структуру для паралельної оптимізації в енергетичних системах, яку дослідники можуть використовувати для систематичного опису своїх досліджень паралелізації та розміщення їх у ландшафті паралельної оптимізації незалежно від області застосування, проблеми, що розглядається, розпаралеленої методології чи використовуваної технології.

Запропонований підхід, зокрема, включає як алгоритмічний дизайн, так і проблеми обчислювальної реалізації паралельної оптимізації, які часто розглядаються окремо в літературі. По-друге, ми використовуємо інтегративну структуру для консолідації попередніх досліджень у галузі енергосистем щодо паралельної оптимізації.

Крім того, існують труднощі з керуванням інформацією та даними через мільйони інтелектуальних лічильників, якими необхідно ефективно керувати. Хмарні обчислення можуть запропонувати більш доступний варіант для аналізу та зберігання даних, як показано на рисунку 1.7.

Нещодавно [7] була представлена структура, яка містить EMS, що зберігається в службі хмарних обчислень, як комбінацію моніторингу різних

джерел живлення, а також керування процесом заряджання та розряджання з урахуванням акумуляторів енергії.

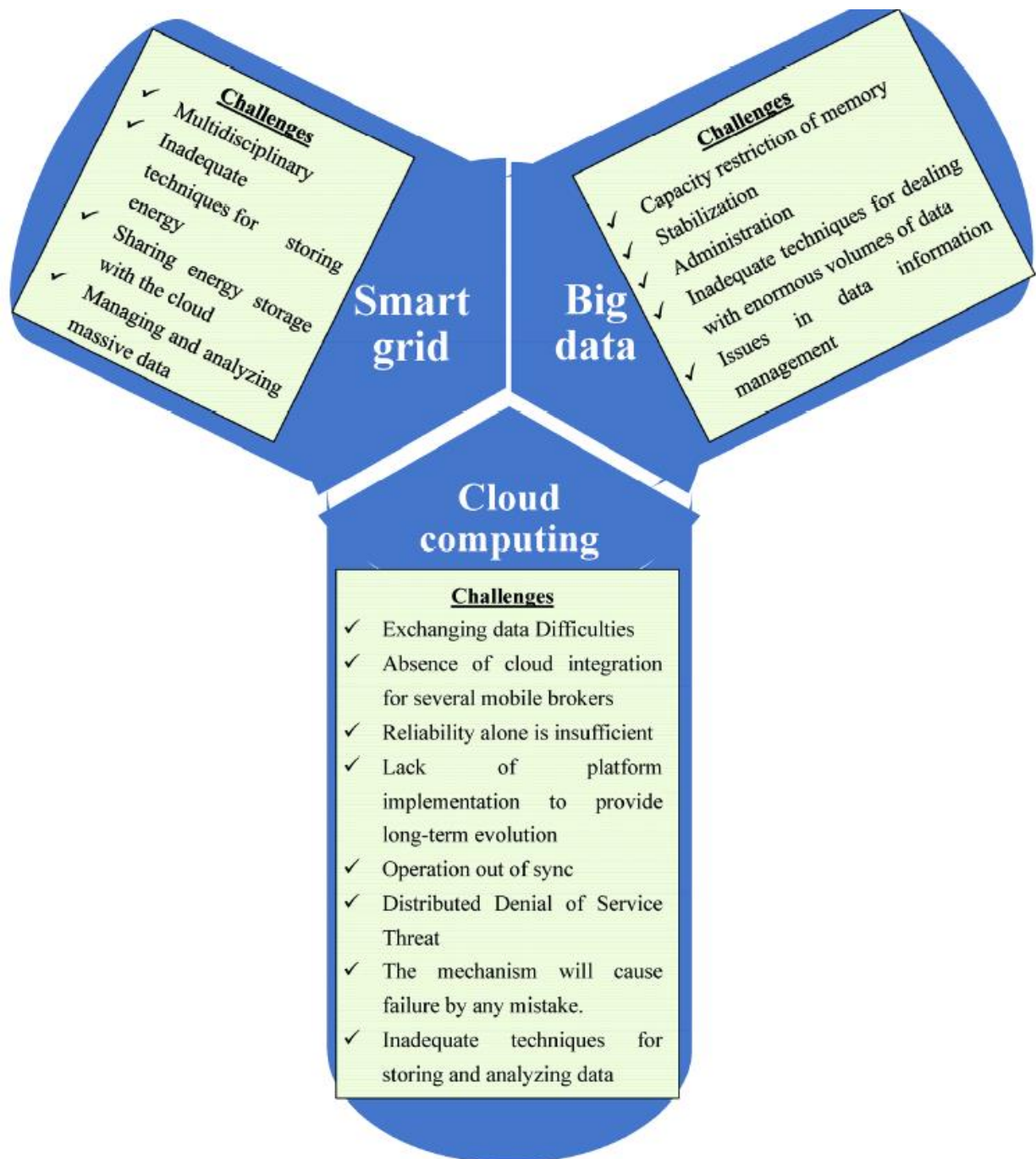


Рисунок 1.7 – Ключові характеристики великих даних у розумних мережах

Ця модель корисна для роботи з найкраще оптимізованою системою та для керування комутаторами концентратора живлення перед тим, як потрібно

працювати з даними, які надходять із системи живлення. Таким чином, це дослідження зосереджено на вирішенні цієї проблеми шляхом використання першого кроку оптимальної системи, якою буде Big Data Analytics з використанням фреймворків на основі хмарних обчислень для систем керування енергоспоживанням через огляд аналітики великих даних із використанням поточних фреймворків хмарних обчислень для систем керування енергоспоживанням.

Ми пропонуємо першу універсально застосовну структуру для паралельної оптимізації в енергосистемах, яку дослідники можуть використовувати для систематичного опису своїх досліджень паралелізації та розміщення їх у ландшафті паралельної оптимізації незалежно від області застосування, проблеми, що розглядається, методології розпаралелювання чи технології використовується. Запропонований підхід, зокрема, включає як алгоритмічний дизайн, так і проблеми обчислювальної реалізації паралельної оптимізації, які часто розглядаються окремо в літературі. По-друге, ми використовуємо інтегративну структуру для консолідації попередніх досліджень у галузі енергосистем щодо паралельної оптимізації. Хмарні обчислення можуть стати чудовим доповненням до будь-якої системи, яка прагне знайти оптимальне рішення, особливо до інтелектуальних систем живлення, які вирішують завдання масивних даних із великомасштабних областей інтелектуальної енергосистеми. Усі ці процеси та дані будуть збережені та керовані структурою хмарних обчислень за допомогою cloudsim.

1.5 Постановка мети та задач роботи

У кваліфікаційній роботі розглядається концепція архітектури хмарних обчислень, яка може відповідати багаторівневим вимогам у режимі реального часу для покращення ефективності обробки великих даних для різних сценаріїв застосування методів управління з моніторингом енергосистеми.

Метою роботи є підвищення ефективності методів управління хмарними системами шляхом розробки архітектурної моделі обробки великих даних з моніторингом енергоспоживання.

Для реалізації мети роботи необхідно вирішити наступні задачі:

- провести аналіз сучасного стану та проблем моніторингу енергосистем живлення георозподілених хмарних центрів обробки даних;
- розробити архітектурну модель обробки великих даних з моніторингом енергоспоживання;
- провести аналіз адаптації існуючих хмарних систем для інтеграції до них розроблених рішень.

Об'єктом досліджень є процес управління ресурсами в хмарних системах.

Предмет досліджень: методи обробки великих даних в розподілених хмарних системах

2 МЕТОДИ ОРГАНІЗАЦІЇ ОБЧИСЛЕНЬ У ХМАРНОМУ СЕРЕДОВИЩІ

2.1 Методи паралельних обчислень

Паралельні обчислення – це обчислювальна модель відносно послідовних обчислень [12]. У цьому режимі кілька обчислювальних процесів виконуються одночасно, що відрізняється від звичайного обчислювального методу, який виконується послідовно [13]. Як показано на рисунку 2.1, користувачі пишуть серію комп'ютерних інструкцій для конкретних проблем, формуючи завдання. У режимі послідовного обчислення ці інструкції можуть виконуватися лише одна за одною одним елементом обробки (PE). Розкладання на кілька наборів інструкцій підзадач, які дозволяють одночасне розв'язання, відображається на кількох PE для виконання. Де PE відноситься до одноядерного центрального процесора (ЦП) або ядра багатоядерного ЦП і є основним апаратним блоком для виконання інструкцій [14].

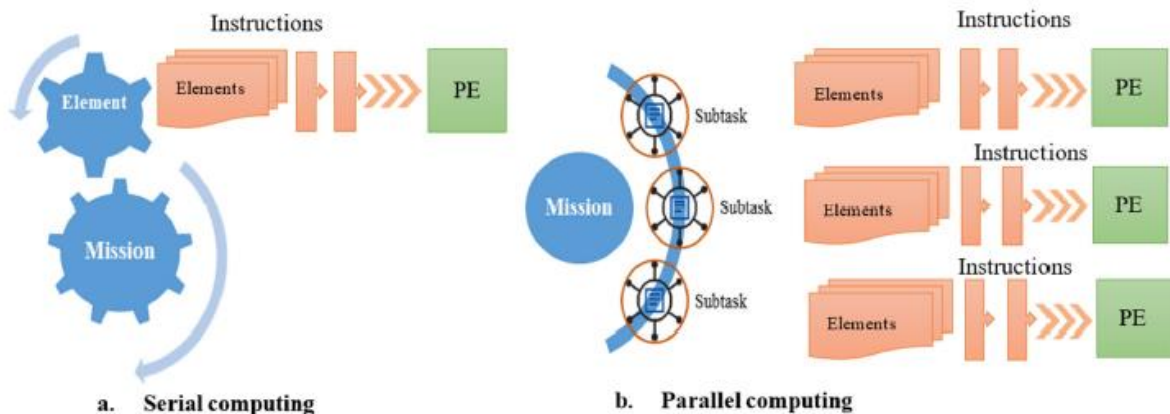


Рисунок 2.1 – Послідовні обчислення та паралельні обчислення

Існує багато методів класифікації технології паралельних обчислень, і нижче наведено лише методи класифікації, пов'язані з цією роботою [95].

Найкласичніша класифікація Флінна – це класифікація з найпростіших інструкцій і методів обробки даних [26,27]. Рисунок 2.2 показує чотири різні типи згідно цієї класифікації:

- SISD є традиційним послідовним методом обчислення: за певного такту, лише одна інструкція виконується і лише один потік даних обробляється;

- SIMD використовує одну інструкцію для обробки кількох потоків даних одночасно за певний такт (сучасні одноядерні комп'ютери також потрапляють у цю категорію та широко використовуються у сферах цифрової обробки сигналів, обробки зображень та обробки мультимедійної інформації);

- MISD використовує кілька потоків інструкцій для обробки одного потоку даних (наразі це лише теоретична модель і не має прикладів застосування);

- MIMD зараз є найпопулярнішим (до цієї категорії належать багатоядерні процесори, які можуть одночасно виконувати декілька потоків інструкцій у кількох різних потоках даних).

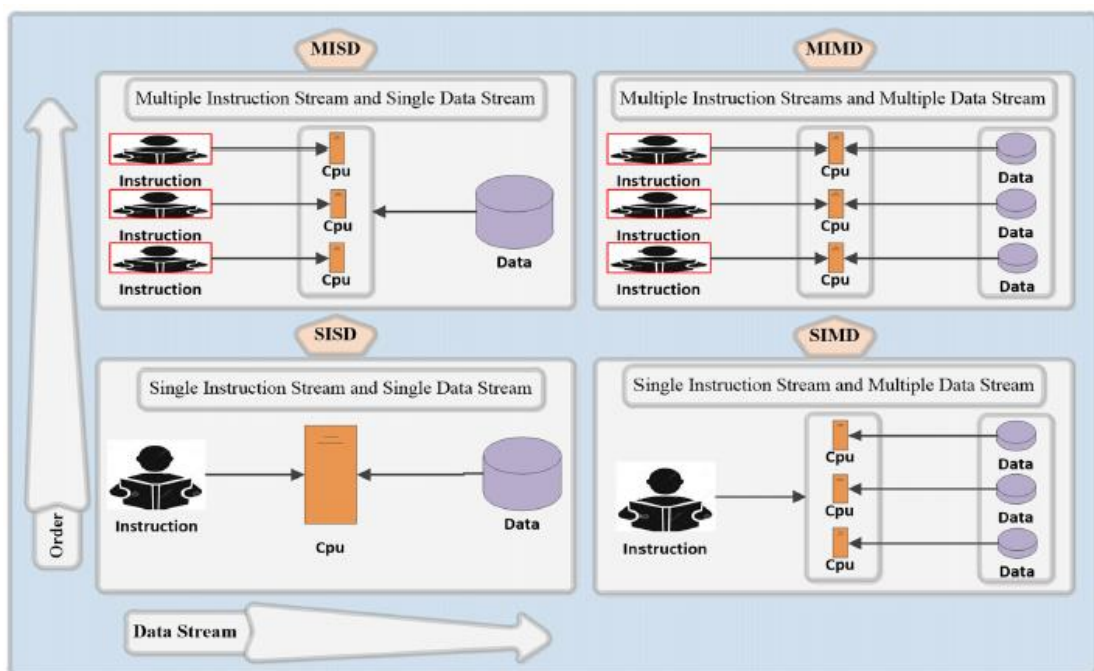


Рисунок 2.2 – Класифікація за Флінном для комп'ютерних систем

Класифікація додатків за обчислювальними характеристиками:

- програми з великим об'ємом даних: такі програми стикаються з величезною кількістю дані з відносно простими обчисленнями [28] (зазвичай паралельні обчислення виконуються шляхом поділу даних, що також можна назвати паралелізмом даних [26,29]);
- застосування з інтенсивним обчисленням: кількість даних, оброблених цим типом застосування не велике, але розрахунок дуже складний [30] (як правило, цей метод декомпозиції завдань використовується для паралельних обчислень, які також можна назвати паралелізмом завдань [31]);
- застосування змішаної інтенсивності: кількість даних і обчислень, які можна використати паралелізм даних або паралелізм завдань для вирішення проблеми, або комбінація двох паралельних методів може бути використана для вирішення проблеми [37].

2.2 Недоліки традиційних паралельних обчислень

Реальні проблеми оптимізації в різних сферах застосування часто є NP-складними, і навіть розробка (мета) евристичних алгоритмів оптимізації може вимагати значних комп'ютерних ресурсів [43]. Зростаюча доступність потужних комп'ютерних можливостей буде використовуватися для вирішення складних проблем оптимізації за допомогою паралельних алгоритмів у багатьох галузях, включаючи фінанси, логістику, виробництво та дизайн [40]. Відсутність уніфікуючих структур для паралельної оптимізації різних методів, секторів застосування та викликів додає гетерогенності [37,38]. Також є проблеми нової інтегративної основи для паралельної обчислювальної оптимізації з використанням оптимізації, методів в різних галузях застосування. Фреймворки поєднує точки зору алгоритмічного проектування та паралельної оптимізації обчислювальної реалізації [44]. Застосування паралельної оптимізації було нерішучим,

оскільки розпаралелювання алгоритмів загалом є складним як з точки зору алгоритмів, так і з обчислювальної точки зору; життєздатною альтернативою алгоритмам розпаралелювання було використання постійного збільшення тактової частоти одиничних процесорів сучасних мікропроцесорів [45].

Ініціативи паралелізації зараз є значно важливішими, ніж це було в минулому завдяки прогресу. Потреба в розпаралелюванні була усвідомлена, і паралельні обчислювальні ресурси стали більш доступними. Основною причиною цього є швидкий розвиток паралельних апаратних архітектур та інфраструктур, включаючи багатоядерні процесори та графічні процесори, локальні високошвидкісні мережі та великі сховища даних, а також бібліотеки та програмні інфраструктури для паралельного програмування [46].

Дослідження виявило велику кількість опублікованих оглядів щодо паралельної оптимізації для конкретних питань, підходів, застосувань, галузей дослідження та технологічних досягнень. Більшість оцінок, які ми виявили, зосереджені на паралельній оптимізації щодо конкретних підходів. Метаевристики домінували в оцінках методологічної літератури, як зазначено.

Не вдалося знайти огляди останніх досліджень паралелізації як точних, так і (мета) евристичних підходів, які були опубліковані в 2008–2017 роках. По-друге, категорії, які використовувалися для визначення та впорядкування ранішої літератури, були різними. Крім цієї неоднорідності, немає жодних структур, які можна було б використовувати для опису паралельної оптимізації різних методів, областей застосування та проблем [47,48]. Це створило фрагментоване загальне уявлення про те, що було досягнуто та що потрібно зробити для паралельної оптимізації в дослідженні операцій. Як побічний ефект, неоднорідність, з якою описуються дослідження розпаралелювання в дослідженні операцій з точки зору алгоритмічного розпаралелювання, обчислювального розпаралелювання та продуктивності розпаралелювання, є високою. Це вигідно з точки зору різноманітності, але

це також створює проблеми, такі як неоднорідність, яка часто забирає багато часу, а в деяких випадках робить неможливим для читачів ідентифікувати вищезгадані характеристики паралельного дослідження, щоб класифікувати дослідження.

Здебільшого в області комп'ютерів, неоднорідність зазвичай відноситься до різних архітектур наборів інструкцій (ISA), так званої комп'ютерної архітектури, або машини, яка виконує команди, визначеними цією ISA, наприклад ЦП [20]. Рівень неоднорідності в сучасних обчислювальних системах поступово зростає, і багато нових процесорів тепер мають інтегровану логіку для взаємодії з іншими пристроями (SATA, PCI, Ethernet, USB, RFID, радіо, UART і контролери пам'яті), а також програмовані функціональні пристрої, блоки та апаратні прискорювачі (графічні процесори, криптографічні співпроцесори, програмовані мережеві процесори, A/V кодери/декодери тощо [21]). Коли система використовує ту саму ISA, але має неоднорідну топологію ЦП, швидкість ядер змінюється [5].

На відміну від звичайних однорідних систем, гетерогенні обчислювальні системи мають різні проблеми [22]. Усі проблеми, пов'язані з однорідними паралельними системами обробки, викликані наявністю багатьох процесорів, і ступінь неоднорідності в системі може спричинити нерівномірність у розробці системи, методах програмування та загальних можливостях системи [23]. Можуть існувати три категорії неоднорідності: перша – це архітектура набору інструкцій (ISA): бінарна несумісність може виникнути через різний дизайн набору інструкцій в обчислювальних елементах. Другий – двійковий інтерфейс програми (ABI): для обчислювальних елементів можуть бути можливі різні інтерпретації пам'яті [24]. Це залежить від архітектури та компілятора, що використовується, і може включати згинання, угоду про виклики та розташування пам'яті. По-третє, інтерфейс прикладного програмування (API): можливо, не всі обчислювальні частини матимуть доступ до всіх ОС і бібліотечних служб одночасно [25].

З точки зору взаємозв'язку, обчислювальні частини можуть бути з'єднані різними способами. У той час як деякі частини гетерогенної системи можуть бути узгодженими з кеш-пам'яттю, підтримка узгодженості та узгодженості може явно вимагатися для інших частин системи [26].

Гетерогенна система може включати процесори з однаковою архітектурою з точки зору продуктивності, але з тонкими змінами в мікроархітектурі, які впливають як на продуктивність, так і на енергоспоживання [27]. Проблеми з передбачуваністю продуктивності, особливо при роботі зі змішаними робочими навантаженнями, іноді можуть бути спричинені асиметрією можливостей у поєднанні з непрозорими моделями програмування та абстракціями операційної системи [28,29].

Хоча розділення даних на однорідних платформах часто легко, було продемонстровано, що завдання є NP-задача для загальної неоднорідної ситуації [30]. Було продемонстровано, що існують ідеальні розділи для невеликої кількості розділів, які повністю балансують навантаження та зменшують обсяг зв'язку [31].

Ресурси паралельних обчислювальних засобів зазвичай являють собою один високопродуктивний комп'ютер або високопродуктивний обчислювальний кластер, незалежно від того, чи є вони спільною архітектурою зберігання чи розподіленою архітектурою зберігання, або вони все ще логічно відображені як єдиний високопродуктивний комп'ютер для користувачів паралельних обчислень [30,32].

Обчислювальні ресурси та ресурси зберігання відносно ізольовані фізично, і обчислювальний блок отримує доступ до даних на пристрої зберігання через шину даних або високошвидкісну мережу. Для інтенсивних обчислювальних програм із малими обсягами даних це не становить проблеми [33]. Зіштовхнувшись зі зростаючими вимогами до додатків із інтенсивним об'ємом даних, часті доступи для читання та запису між обчислювальними блоками та блоками зберігання стануть вузьким місцем продуктивності всієї паралельної системи. Ця проблема особливо вірна в

архітектурі спільного зберігання, де дані зберігаються централізовано [34]. Щоб зменшити обсяг даних, що передаються між процесами, кожен працівник має локальну резервну копію всіх даних, які потрібно проаналізувати, і лише метадані даних надсилаються під час призначення завдань [35]. Однак ця попередня умова нелегко задовольнити в кластерах розподілених архітектур зберігання, тому паралельна схема підходить лише для багатоядерних або багатоядерних комп'ютерів при вирішенні проблем, що містять інтенсивні дані [36].

Кластери високопродуктивних комп'ютерів (HPC) складаються з високопродуктивного апаратного забезпечення та є дуже дорогими [37]. Розширення кластера зазвичай використовує метод вертикального розширення, який покращує обчислювальну продуктивність кластера шляхом заміни ЦП, розширення пам'яті та додавання дисків. Однак вертикальне масштабування дуже обмежене, його легко досягти, а оновлення коштує дорого. Масштабованість може бути досягнута горизонтальним розширенням, таким як кластер робочих станцій (COW), але все ще існують інші проблеми в традиційних паралельних обчисленнях на основі цього обладнання [38]. Хоча існуючі традиційні моделі програмування паралельних обчислень, такі як інтерфейс передачі повідомлень (MPI) і OpenMP, були інкапсульовані в нижній частині, і це управління зберіганням даних, розподіл даних, розподіл завдань і планування, синхронізація даних і передача даних, відмовостійкість і багато інших технічні деталі мають обробляти самі користувачі, що все ще дуже громіздко [39]. Користувачі заплутані в багатьох основних технічних деталях, розглядаючи саму проблему програми, що робить паралельне програмування непростим [5].

2.3 Хмарні обчислення

Концепція хмарних обчислень визначена Національним інститутом стандартів і технологій США (NIST): як модель, яка може досягти зручності

для отримання необхідних ресурсів (включаючи мережі, сервери, сховище, програми та послуги) через доступ до мережі на попит, і необхідні ресурси можуть бути швидко надані або вивільнені, з невеликими зусиллями керівництва або невеликою взаємодією з постачальниками послуг [40,41].

Вважається, що хмарні обчислення мають наступні характеристики.

Віртуалізація: віртуалізація є основною технологією хмарних обчислень і багатьма іншими функціями, які залежать від неї. Застосування технології віртуалізації може інтегрувати різні обчислювальні ресурси для формування пулу ресурсів для доступу користувачів [42].

Сервіс-орієнтований: хмарні обчислення надають три рівні послуг, а саме інфраструктуру як послугу (IaaS), платформу як послугу (PaaS) і програмне забезпечення як послугу (SaaS). IaaS — це служба найнижчого рівня, яка безпосередньо надає обчислювальну техніку, пам'ять і мережеве обладнання. Користувачі мають найбільший ступінь свободи та можуть створювати власні платформи та програмне забезпечення. PaaS на один рівень вище, ніж IaaS, надає готову хмарну платформу, заощаджуючи роботу з розробки платформи. SaaS забезпечує більш зручні послуги; користувачі можуть безпосередньо використовувати надане програмне забезпечення без будь-якої розробки [43].

Еластичність і масштабованість: масштаб хмари можна легко розширити, не впливаючи на хмарні служби, які наразі надаються зовні. Ресурси в хмарі безмежно бажані для користувачів, їх можна автоматично надавати та швидко повертати на вимогу [44].

Надійність і універсальність: технологія хмарних обчислень надає різноманітні стійкі до збоїв механізми для забезпечення високої надійності послуг [45]. Дані розміщуються в кількох копіях, щоб запобігти втраті даних через збій обладнання [46]. Обчислювальні служби, які були зупинені через апаратні збої, все ще можуть продовжуватися в іншому місці через міграцію віртуальної машини. Віртуалізація робить ресурси хмарних обчислень

прозорими для користувачів і підтримує додатки в різних галузях одночасно [47];

Економія на масштабі: платформа хмарних обчислень не має високих вимог до апаратного забезпечення, і велика кількість неактивних звичайних комп'ютерів може бути інтегрована в пул ресурсів за допомогою віртуалізації [33]. Для користувачів це заощаджує витрати на апаратне забезпечення та щоденні витрати на керування самостійно створеними платформами [57]. Для постачальників хмарних послуг універсальність хмарних обчислень значно покращила використання ресурсів, а масштаб значно збільшив економічні вигоди [48].

2.4 Середовище хмарних обчислень

Розумна мережа – це неоднорідне та складне середовище, що містить різні види пристроїв, мереж, систем і даних. IEC 61970 і IEC 61850 були відкриті в результаті дослідження основних стандартів Smart Grid і є платформами з відкритим кодом на основі хмарних обчислень. IEC 61970 визначає інтерфейс прикладної програми (API), тоді як IEC 61850 визначає інтерфейс абстрактних комунікаційних послуг (ACSI) [49].

Сумісність зі стандартами IEC61970 і IEC61850 для технологій хмарних обчислень, таких як Hadoop, Spark і Storm, передбачає ретельний аналіз існуючих систем, переклад даних, прийняття форматів даних, сумісних зі стандартами, інтеграцію систем і ретельне тестування, щоб гарантувати, що системи функціонують за призначенням. Оскільки ці два стандарти визначають моделі даних та інтерфейс окремо, моделі не є однорідними, і безперервний зв'язок між підстанцією та центром керування неможливий. У той час як IEC 61970 визначає потужність інформаційної моделі та широко використовується в корпоративній інтеграції. IEC 61850 обмежується обміном даними всередині обладнання підстанції. Дослідження показали, що Hadoop може об'єднати ресурси неактивної системи живлення

та забезпечити «суперобчислювальну здатність» для платформи інтеграції даних інтелектуальної мережі.

Платформа підтримки системи диспетчеризації електромережі та прикладне програмне забезпечення повинні бути оновлені відповідно до специфікації інтерфейсу компонентів (CIS) і стандартів Загальних інформаційних моделей (CIM). Дані інтегруються за стандартом IEC61970 і підключаються до інтелектуальної мережі через платформу для обміну даними [50]. Основним користувачем цих стандартів буде не людина, а комп'ютер, і вони мають бути машинозчитуваними. Водночас це дуже складні документи, що містять тисячі різних елементів. Повна серія стандартів IEC 61850 тепер доступна як глобальний пакет. Вони видаються з доступними пов'язаними компонентами коду. Серія включає не менше 35 документів, що стосуються автоматизації підстанцій, інтеграції DER або кібербезпеки, і це лише деякі [51]. З розвитком технології хмарних обчислень і потребами обробки великих даних з'явилася низка нових обчислювальних моделей і моделей програмування, які надають користувачам базову платформу для паралельного програмування в хмарному середовищі та максимально захищають користувачів від нижній шар. Деталі представлені користувачеві через абстрактний інтерфейс вищого рівня [52].

2.4.1 Технологія Hadoop

Hadoop – це масивна система обробки даних з відкритим вихідним кодом від Apache Software Foundation. Він містить багато компонентів для зберігання або обробки, таких як розподілена файлова система Hadoop (HDFS) і модель паралельних обчислень MapReduce [53]. HDFS має характеристики розподіленого зберігання, високий рівень одночасного доступу, високий рівень помилок толерантність, проста узгодженість і забезпечує надійне середовище зберігання для моделей паралельних обчислень [44]. Він використовує архітектуру «головний-підлеглий» і

побудований на фізичному кластері, об'єднаному кількома комп'ютерами через мережу. Нижній рівень – це локальна файлова система операційної системи. Його архітектура показана на рисунку 2.3, показує, що HDFS включає головний вузол NameNode, резервний головний вузол SecondaryNameNode і набір підлеглих вузлів DataNode. NameNode відповідає за керування простором імен HDFS, збереження всіх метаданих і відповіді на запити клієнтського доступу. SecondaryNameNode використовується для вирішення проблеми єдиної точки відмови Hadoop. DataNode відповідає за фактичне зберігання та керування надлишковими блоками даних файлів, а файли блоків даних фактично зберігаються в локальній файлової системі кожного вузла [45].

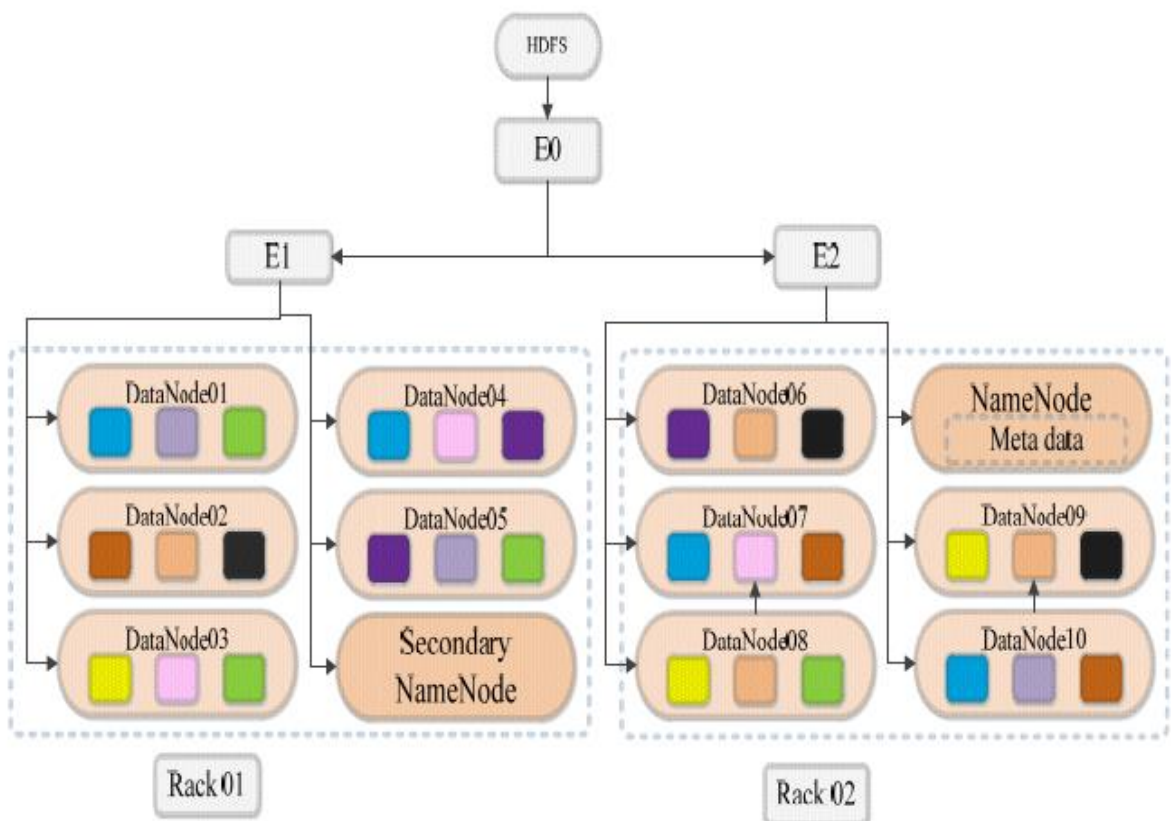


Рисунок 2.3 – Архітектура HDFS

Основна ідея MapReduce в основному така ж, як і традиційних паралельних обчислювальних моделей, таких як MPI, яка полягає у «розділяй

і володарюю» велику кількість даних. Система забезпечує два простіших інтерфейси, Map і Reduce, які автоматично виконують багато основних функцій, таких як розподіл завдань і планування, зв'язок, балансування навантаження та відновлення після збоїв [15]. Рисунок 2.4, показує режим програмування MapReduce [14]. Вхідний файл поділяється на кілька фрагментів (InputSplit) відповідно до певного формату та перетворюється на пари ключ-значення <ключ, значення>, які вводяться в Mapper для обчислення, а проміжними результатами є пари ключ-значення. Збірка агрегується Reducer після фази перемішування, а результат зберігається в HDFS.

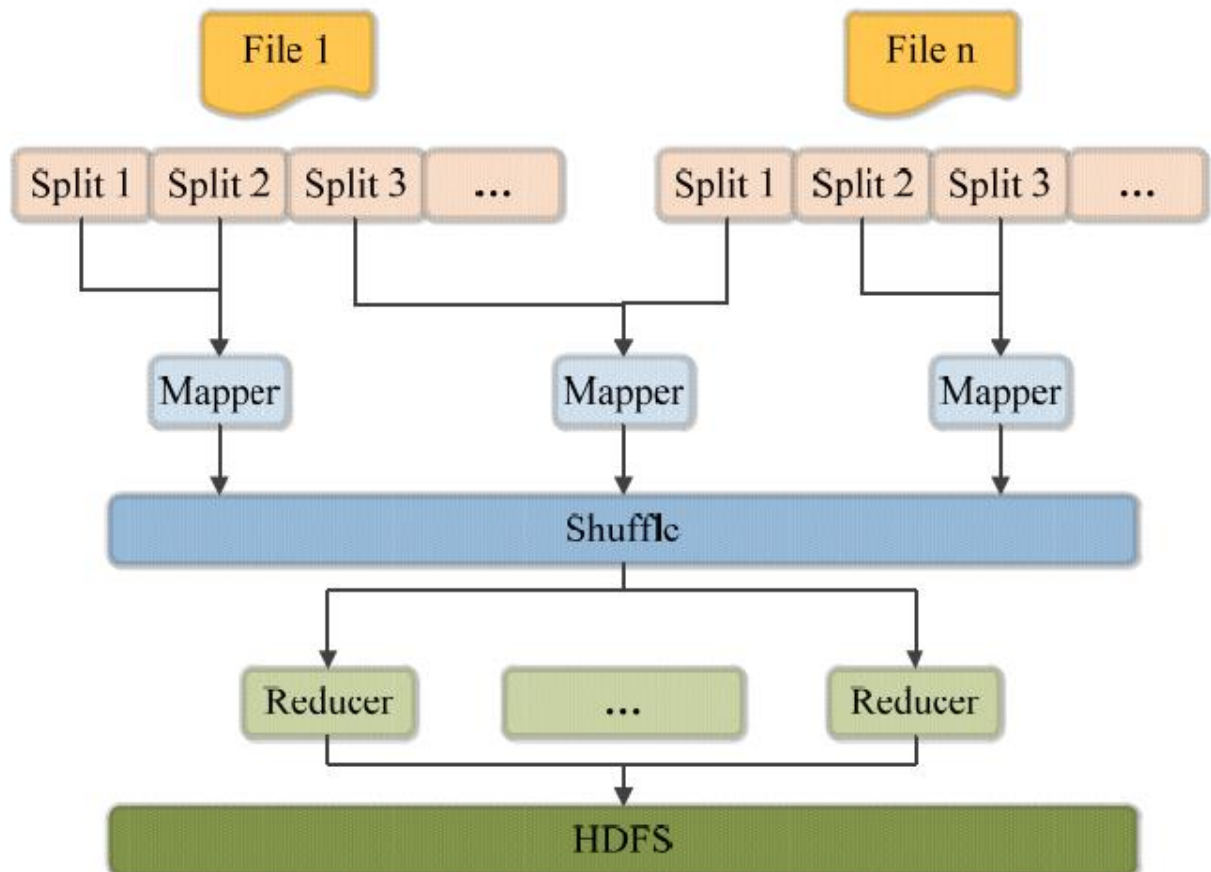


Рисунок 2.4 – Модель програмування MapReduce.

Nadoop MapReduce в основному орієнтований на пакетний режим масивних статичних даних, і його продуктивність у реальному часі не висока

[41]. Збірка агрегується Reducer після фази перемішування, а результат зберігається HDFS. Hadoop MapReduce в основному орієнтований на пакетний режим масивних статичних даних, і його продуктивність у реальному часі не висока [42]. Тому він надто швидко обробляє та аналізує масивні історичні дані, накопичені під час моніторингу стану енергетичного обладнання, сподіваючись отримати з них цінні знання [43,4].

2.4.2 Технологія Spark

Spark – це фреймворк паралельних обчислень загального призначення, схожий на Hadoop MapReduce, який з'явився в 2012 році [7]. Різниця полягає в тому, що він поміщає дані (включно з деякими проміжними даними) у пам'ять для обчислення, уникаючи великого обсягу дискового введення/виведення, спричиненого частим читанням і записом HDFS під час процесу обчислення. Таким чином, spark підходить для ітеративних та інтерактивних обчислювальних сценаріїв і навіть для загальних прикладних сценаріїв, які ефективніші, ніж Hadoop MapReduce [15]. Функції Spark для обчислення в пам'яті з його базової абстракції Resilient Distributed Dataset (RDD) [15]. RDD є лише для читання, відмовостійкими, розподіленими обчисленнями, роздільними, грубими перетвореннями та зберіганням у пам'яті. Кожен розділ нового RDD, створений під час процесу обчислення Spark, має зв'язок залежності з розділом його батьківського RDD завдяки обчисленню, який називається лінійним походженням. Втрачений розділ RDD можна відновити з попереднього RDD шляхом відстеження походження, щоб реалізувати відмовостійкість. У той же час Spark розділяє весь обчислювальний процес на кілька етапів відповідно до різних залежностей між RDD, кожен етап генерує завдання та створює завдання в одиницях розділів RDD і розподіляє їх на кілька етапів. Рисунок 2.5, показує процес паралельно на двох обчислювальних вузлах, а обчислювальна модель Spark [16]. Обчислювальна модель багатша та гнучкіша, ніж єдина модель

MapReduce Hadoop, і сумісна з різними джерелами даних, такими як HDFS, HBase та Hive [16]. В даний час Spark широко використовується в таких інтернет-компаніях, як Amazon, Yahoo і Taobao [162], і він все ще знаходиться на стадії дослідження в електроенергетиці, і дослідження все ще потрібно проводити в поєднанні з типовими сценаріями застосування [13]. Заснований на Spark, не в змозі зрозуміти, як складний алгоритм обробки сигналу може виконувати швидкі обчислення, коли обсяг даних великий, що компенсує прикладні сценарії, які Hadoop MapReduce [14].

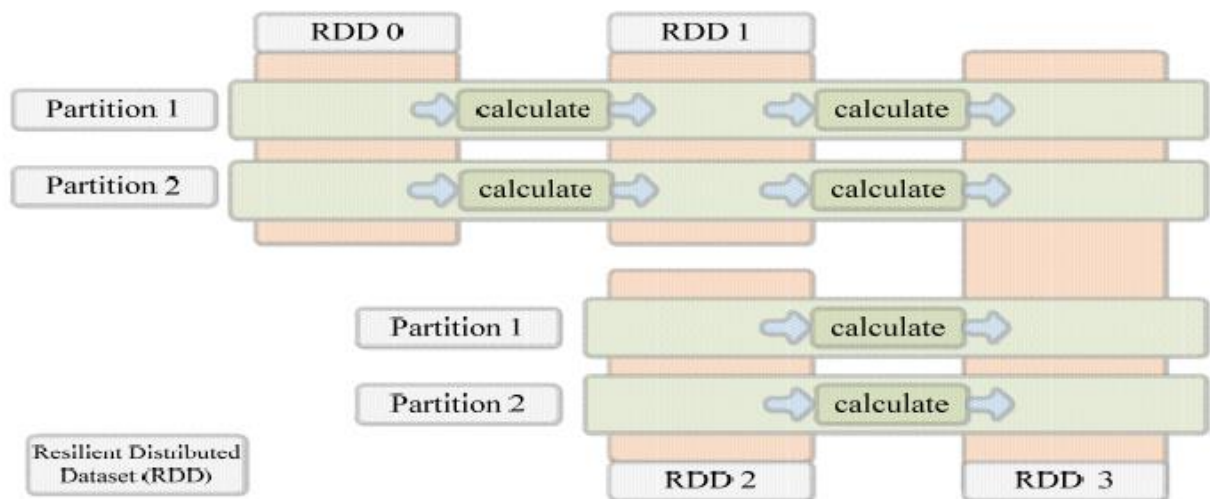


Рисунок 2.5 – Обчислювальна модель на основі стійкого розподіленого набору даних (RDD)

2.4.3 Технологія Storm

Storm насамперед орієнтований на аналітику в реальному часі для великих безперервних потоків даних, на відміну від Hadoop, який зосереджується на пакетній обробці величезних обсягів даних. Хоча Spark Streaming також може виконувати функцію потокового обчислення шляхом декомпозиції пакетних завдань, його затримка довша, ніж у Storm [16].

Storm також приймає архітектуру masterslave і використовує ZooKeeper для координації всього кластера. Де головний вузол називається вузлом керування та запускає демон Nimbus, який відповідає за публікацію програм топології, розподіл завдань і моніторинг стану кластера [16]. Підлеглий вузол називається робочим вузлом, який запускає демон Supervisor, який відповідає за прийняття призначених завдань і запуск робочого процесу віртуальної машини Java (JVM), що показано на малюнку.12а [16,18].

Рисунок 2.6 показує топологію, де Spout є записом даних топології, підключається до зовнішнього джерела даних і перетворює дані в кортежі, які надсилаються до Bolt. Логіка обробки для кортежів інкапсульована в Bolts, і після завершення обробки кортежі можуть бути передані наступним Bolts [16].

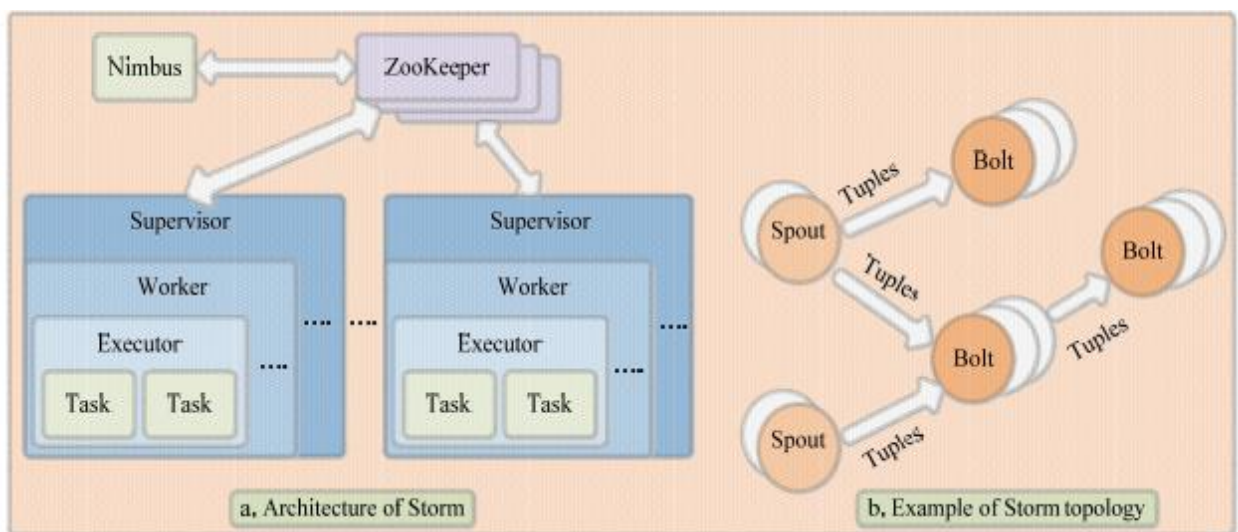


Рисунок 2.6 – Архітектура і топологія Storm

Компоненти Spout і Bolt пов'язані стратегією групування потоків і можуть бути налаштовані як кілька екземплярів для досягнення паралельної обробки. Кожен екземпляр зрештою сформує завдання, яке буде заплановано для виконання [17].

3 ЗАСТОСУВАННЯ ХМАРНИХ ОБЧИСЛЕНЬ ДЛЯ УПРАВЛІННЯ ЕНЕРГОСПОЖИВАННЯМ ПРИ ОБРОБЦІ ВЕЛИКИХ ДАНИХ

3.1 Порівняння паралельних обчислень із хмарними обчисленнями

Хмарні обчислення – це злиття багатьох технологій, у тому числі технології паралельних обчислень, тому вони не еквівалентні паралельним обчисленням, а їх зміст багатший [36,41].

Паралельні обчислення зазвичай стосуються спеціально розробленого паралельного комп'ютера, тоді як хмарні обчислення об'єднують декілька звичайних комп'ютерів для досягнення мети покращення продуктивності обчислень [12,13]. У широкому розумінні обчислювальна технологія, яку він приймає, також належить до паралельних обчислень. Технічний аспект паралельних обчислень зосереджується лише на обчисленнях і ігнорує зберігання даних [14]. Це пояснюється тим, що традиційні моделі паралельного програмування, такі як MPI, розроблені для високопродуктивних обчислень і невеликої кількості даних [14].

Хмарні обчислення включають сховище та обчислення, і вони співпрацюють один з одним. Наприклад, у системі Hadoop дані зберігаються розподіленим способом, а потім обчислення переміщуються до місця розташування даних для виконання, оскільки мобільні обчислення ефективніші, ніж переміщення даних [15]. З точки зору відповідних галузей, паралельні обчислення підходять для наукової обчислювальної сфери з вимогами до високопродуктивних обчислень [2]. Він орієнтований на додатки з інтенсивним обчисленням і вимагає від користувачів високої професійної якості, щоб мати змогу працювати з багатьма деталями низького рівня [10]. Хмарні обчислення надають послуги користувачам на трьох різних рівнях, що простіше для користувачів. Хмарні обчислення є ключовою технологією для обробки великих даних і підходять для додатків

із інтенсивним об'ємом даних, але великі витрати на управління системою та технічне обслуговування роблять її непридатною для додатків із інтенсивним обчисленням із невеликим обсягом даних [25]. Підводячи підсумок, можна сказати, що технології паралельних обчислень і хмарних обчислень мають широкий спектр застосувань у різних сферах, включаючи енергетику [39,46]. Вони доповнюють один одного, а не виключають один одного, і кожен має різні сценарії застосування [17].

3.2 Розподілені хмарні обчислення та паралельні обчислення

Розподілені хмарні обчислення – це новий метод з'єднання даних і додатків, що надходять із кількох місць. Спільний ресурс, географічно розподілений між кількома користувачами або системами, називається розподіленим. Здатність виконувати кілька завдань одночасно є ключовою характеристикою хмарних обчислень, яка також спрямована на зменшення споживання ЦП, скорочення часу перемикавання, скорочення часу очікування для обробки даних, підвищення пропускну здатності сервера та покращення обробки даних і швидкості зв'язку. Інша функція спрощує використання будь-якої хмарної програми для спілкування з користувачами в різних місцях. Останнім важливим компонентом є підвищення продуктивності сервера, оскільки продуктивність зв'язку має вирішальне значення.

Після ознайомлення з кожною посиланням, включеною в табл3, було визначено, що [18] була найкращою роботою з розподілених хмарних обчислень, оскільки охоплювала більшість функцій.

Три різні методи паралельної обробки: розподілена, спільна та гібридна системи пам'яті. Фокус розподілених паралельних обчислень визначив певні ключові критерії. Важливою функцією, згаданою в кількох ресурсах, є підвищення продуктивності за допомогою техніки балансування навантаження шляхом розподілу процесу та створення балансу між серверами для обробки завдань і підвищення продуктивності нашої

розподіленої системи. Іншою функцією є мінімізація витрат на ресурси, тому що коли ми розподіляємо навантаження між серверами, ми можемо мінімізувати витрати на ресурси, такі як ЦП, пам'ять і сховище. Оскільки бажано використовувати систему для обробки запитів користувача з низьким часом відповіді, усі посилання застосовують цей принцип, рекомендуючи метод для розподілених паралельних обчислень на основі цього фактора. Після перегляду посилань у цій роботі було визначено, що метод [18] був кращим, оскільки він забезпечує широкий спектр характеристик, таких як балансування навантаження, підвищення продуктивності системи та зниження часу реакції та витрат на ресурси.

3.3 Основи застосування хмарних обчислень в енергетичній системі

На початку існування Інтернету вартість обладнання була відносно високою. З безперервним розвитком інформаційних і мережевих технологій дані, які генеруються й обробляються в Інтернеті, зросли в геометричній прогресії. Щоб впоратися з цими змінами, інвестиції в апаратне обладнання мають збільшити вхідні витрати [38]. Однак, незважаючи на високу вартість інвестицій у апаратне забезпечення, масштабованість системи дуже погана, ефективність передачі інформації низька, а різниця в продуктивності між пристроями також є поганим ефектом обробки [17]. Теоретичною основою хмарних обчислень є віртуалізація обладнання та послуг, а потім з'єднання кількох розподілених вузлів на хмарній платформі для реалізації суперпозиції та з'єднання обчислювальних ресурсів [19]. Щоб краще завершити обробку масивних даних, розподіліть велику кількість завдань обробки між віртуальними вузлами в пулі ресурсів [29,36]. Хмара – це дуже широке поняття з багатьма типами, і зазвичай його класифікують відповідно до різних об'єктів і типів послуг [39].

Публічна хмара. Як випливає з назви, це хмарний сервіс, відкритий для громадськості. Це широкомасштабний, недорогий і найпопулярніший

хмарний сервіс для громадськості. Найбільш типовою програмою є Amazon Web Services («AWS»). Додаток надає клієнтам по всьому світу повний набір інфраструктурних і хмарних рішень. AWS надає користувачам повний набір послуг хмарних обчислень, які можуть допомогти підприємствам зменшити витрати на інвестиції в ІТ та витрати на технічне обслуговування та легко перейти до хмари [13].

Приватна хмара. Це хмара, яка не надає публічних послуг і використовується групою чи організацією. Надавати внутрішнім користувачам послуги приватної хмари. Оскільки вони не можуть використовуватися публічно, більшість брендмауерів налаштовано [14]. Типовим представником приватної хмари є план Blue Cloud, запущений IBM. Blue Cloud базується на відкритих стандартах і програмному забезпеченні з відкритим кодом на основі програмного забезпечення, системних технологій і послуг IBM [15]. Синя хмара, розроблена понад 200 дослідниками IBM у всьому світі, допоможе клієнтам швидко та легко вивчити інфраструктуру хмарних обчислень для екстремальних обчислень [16].

Гібридна хмара. Тобто поєднання загальнодоступної та приватної хмари є між приватним і загальнодоступним, як-от віртуальна приватна хмара (VPC) Amazon [53]. VPC – це динамічно наданий пул загальнодоступних хмарних обчислювальних ресурсів, який вимагає використання протоколів шифрування, протоколів тунелювання та інших процедур безпеки для передачі даних між приватними підприємствами та постачальниками хмарних послуг [19–20]. Послуги, які надає кожен рівень, такі.

Прикладний рівень надає користувачам різноманітне прикладне програмне забезпечення та послуги, необхідні для зручного інтерфейсу [21]. Прикладний рівень безпосередньо відповідає потребам клієнтів і надає корпоративним клієнтам корпоративні програми, такі як планування ресурсів підприємства (ERP) і управління взаємовідносинами з клієнтами (CRM) [22] та автоматизації офісу (OA) [23].

Рівень платформи надає послуги для користувачів, які можуть використовувати платформу для реалізації цінності, яку вони хочуть досягти [24].

Рівень інфраструктури надає послуги на рівні інфраструктури, тобто створення інфраструктури платформи хмарних обчислень безпосередньо відкрито для користувачів, щоб вони могли використовувати потужні можливості зберігання та обчислення хмарних обчислень.

Користувачі можуть безпосередньо зберігати файли та виконувати обчислення в хмарі, а також інфраструктуру можна розподіляти незалежно, що еквівалентно користувачеві, який має масштабований комп'ютер із великим простором для зберігання та суперкомп'ютерною продуктивністю через термінал [25].

Загалом, перевага хмарних обчислень полягає в тому, що вони можуть інтегрувати різні ресурси без особливих вимог до цих обчислювальних ресурсів і не потребують конкретного комп'ютера з високою продуктивністю [33]. Користувачі можуть легко отримувати хмарні сервіси через звичайні термінали та використовувати можливості суперкомп'ютерів. Таким же чином інтелектуальна хмарна система також може бути побудована в енергосистемі. Користувачі можуть контролювати та відстежувати систему живлення безпосередньо на мобільному терміналі, щоб покращити зручність системи живлення [26–28].

В даний час для побудови інтелектуальної мережі структурна схема показана на рисунку 3.1. У порівнянні з попередньою моделлю обчислень хмарні обчислення зробили багато якісних проривів. Його масштаб великий, але він має високу надійність [29]. Тип розширення, який об'єднує ці функції разом, робить його неперевершеними перевагами [21]. Google є ініціатором ідей хмарних обчислень. Пізніше на основі попередньої роботи Google розробив важливу хмарну обчислювальну платформу з відкритим кодом Hadoop. Ключ Hadoop включає дві основні системи: розподілену файлову систему HDFS і структура розподілених обчислень [21,23].

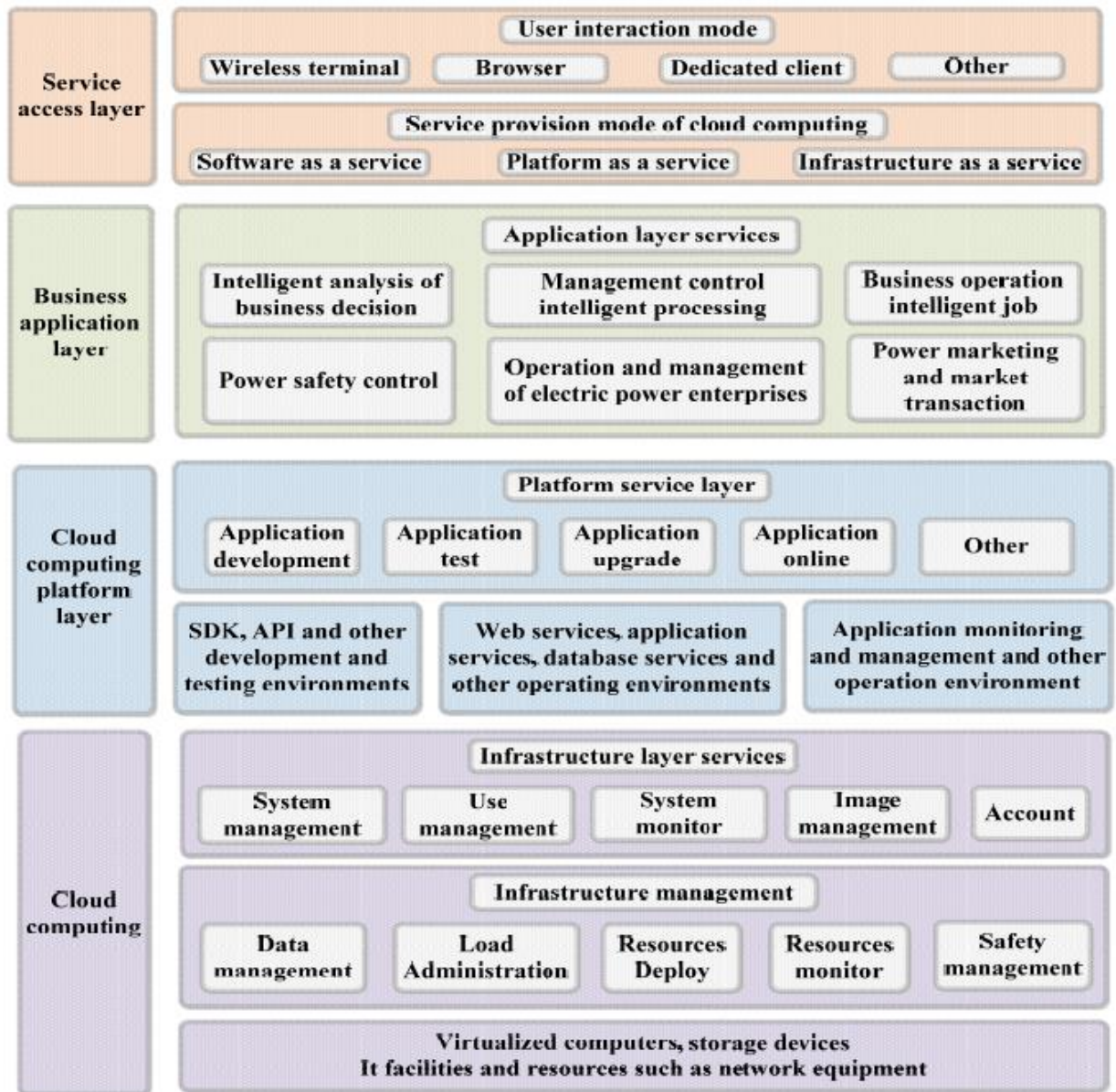


Рисунок 3.1 – Структура грид-інтелектуальної інформаційної платформи

Хмарні інструменти та технології для керування мікромережами — це набір програмних і апаратних рішень, призначених для моніторингу та керування використанням електроенергії та розподілом мікромереж. Архітектура хмарних обчислень забезпечує надійне й ефективне керування даними, обчислювальними ресурсами та додатками з такими функціями, як хмарні обчислення з ізоляцією несправностей, що означає здатність системи ізолювати збій або несправність в одній частині системи, щоб вона не впливає на всю систему. Крім того, самовідновлення системи стосується

здатності системи хмарних обчислень автоматично виявляти та відновлюватися після збоїв без втручання людини. У хмарному обчислювальному середовищі керування мікромережею можна досягти за допомогою хмарних інструментів для моніторингу та контролю споживання та розподілу електроенергії в мікромережі. Ці системи можуть надавати дані в режимі реального часу та уявлення про споживання енергії окремими пристроями та системами, дозволяючи ефективніше та ефективніше керувати мікромережею. Взаємодія з локальними даними може бути досягнута за допомогою API (інтерфейсів прикладного програмування) та інші протоколи зв'язку, які забезпечують бездоганну інтеграцію між хмарою та локальною мережею системи. API можна використовувати для доступу до хмарних ресурсів і сервісів, а також для обміну даними між хмарою та локальними системами. Це дозволяє ефективніше та ефективніше використовувати ресурси та забезпечує зручний спосіб доступу до хмарних служб із локальних систем.

У майбутньому хмарні обчислення енергетичних систем матимуть широке застосування, яке можна описати як «майбутнє можна очікувати», як показано на рисунку 3.2. Його ефективне застосування може забезпечити велику кількість послуг з високою доданою вартістю всередині та за межами галузі та має високу практичну цінність для прибутковості та контролю рівня витрат енергетичних компаній [24].

Модель підсистеми має довгострокові переваги для національної оборони, військового та медичного обслуговування. Цінність застосування лише в енергосистемі є незмірною.

По-перше, це допоможе мережевим компаніям здійснювати моніторинг експлуатації та технічного обслуговування мережі та підвищити чутливість реагування [25]. Використовую дані, зібрані з енергосистеми, щоб відстежувати, контролювати або коригувати генерацію електроенергії, навантаження та статус несправності в мережі та відповідним чином реагувати на помилку або оновлення в електромережі [29,36].

По-друге, це допоможе мережевим компаніям проводити спеціальний аналіз технічного обслуговування, експлуатації та технічного обслуговування обладнання, підвищити надійність системи, рівень кваліфікації електропостачання, зменшити витрати та зменшити відключення електроенергії [27]. У сфері технічного обслуговування, експлуатації та технічного обслуговування електромереж, шляхом вибору ключових показників енергетичного обладнання з трьох аспектів безпеки, вигоди та вартості, аналіз взаємного впливу «безпека», «вигода» та «вартість» в управлінні техобслуговуванням координація цих трьох факторів всебічно оптимізується, і в той же час реалізується онлайн-моніторинг показників технічного обслуговування електромережеских підприємств у режимі реального часу, що забезпечує керівництво та послуги для формування стратегії технічного обслуговування компанії [28].

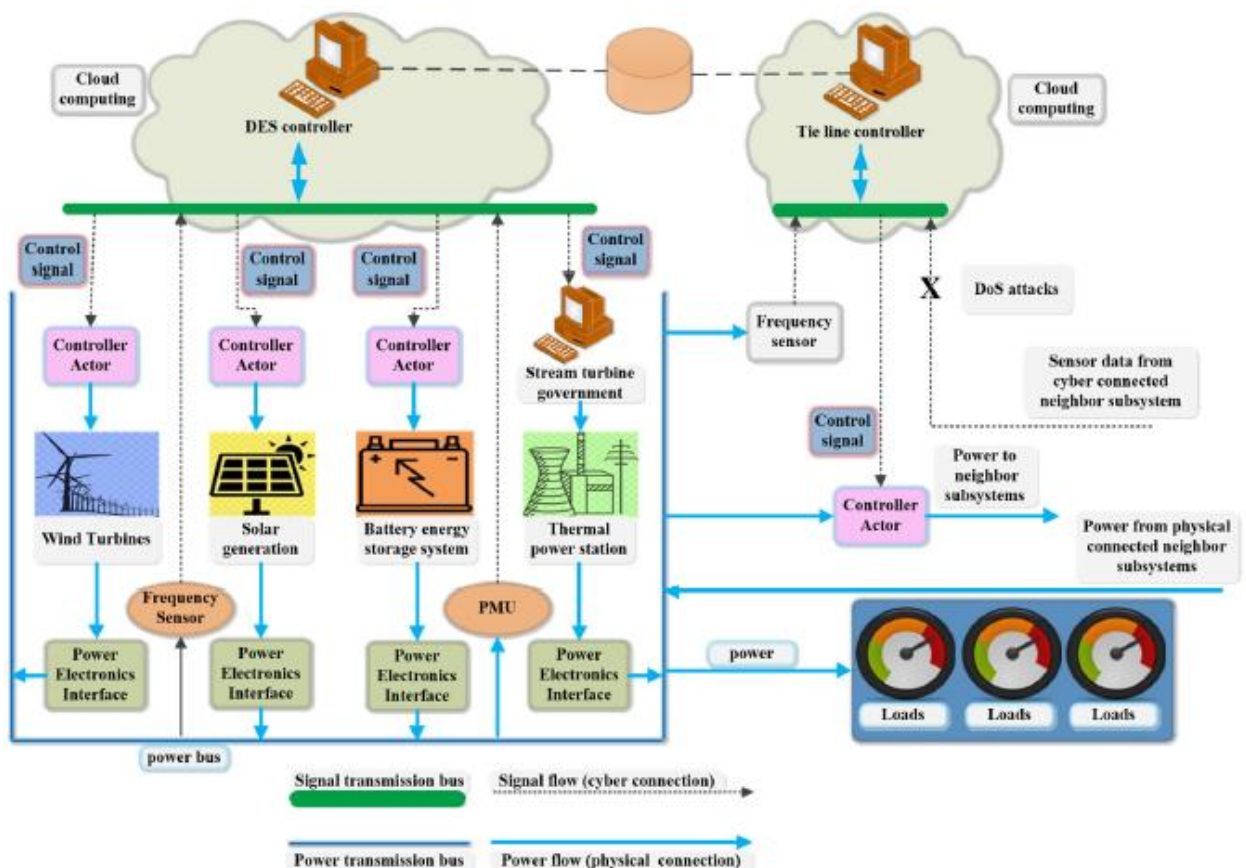


Рисунок 3.2 – Принципова схема системи

Деякі моменти, які потребують покращення через обмеження попередньої роботи, це особисті здібності та джерело даних:

- ємність даних ще недостатньо велика, а інтервал зміни розмірності даних малий [21]; дані про електроенергію надходять із споживання електроенергії в житлових приміщеннях і не включають дані про споживання електроенергії в інших сферах, таких як промислова електроенергія [22];

- сфера розробки паралельних алгоритмів відносно вузька, а область застосування в обробці даних енергосистеми недостатньо широка; однак із зростаючою інформатизацією енергосистеми та безперервною кількісною оцінкою даних про потужність сфера застосування технології інтелектуального аналізу даних продовжує розширюватися;

- паралельні алгоритми можуть бути розроблені в більшій кількості аспектів, щоб посилити ефект обробки даних і здійснювати виробництво й диспетчеризацію всієї енергосистеми [23, 24]; розробити комплексну архітектуру платформи хмарних обчислень на основі технології хмарних обчислень і обробки великих даних, яка забезпечує еталонний план побудови обчислювальної платформи центру моніторингу енергетичного обладнання [25,26];

- щоб покращити швидкість реагування в режимі реального часу на дані онлайн-моніторингу енергетичного обладнання, потрібна структура обробки в реальному часі для потокових даних на основі Storm, а також метод класифікації моделі інкрементального прогнозування змінних [27–29].

ВИСНОВКИ

В кваліфікаційній роботі розглянуті основні концепції та моделі програмування паралельних обчислень і проаналізовані недоліки традиційних паралельних обчислень для поточних сценаріїв великих даних на конкретному прикладі застосування. Представлено рішення хмарних обчислень для моніторингу стану енергетичного обладнання на фоні великих даних, представлено концепції та характеристики хмарних обчислень, а також коротко описано нові моделі паралельного програмування, такі як Hadoop, Spark і Storm. . Нарешті, для різних сценаріїв застосування в області моніторингу стану енергетичного обладнання комплексна архітектура платформи хмарних обчислень розроблена, щоб задовольнити багаторівневі вимоги в режимі реального часу, спираючись на досвід Інтернет-сфери.

Існує потреба у вдосконаленні алгоритму паралельного виявлення для ефективності виявлення аномалій силового навантаження. Виявлення аномального значення є лише першим кроком в аналізі аномального значення потужності. Після виявлення аномального значення можна швидко проаналізувати причину аномальних даних, наприклад економічні причини, регулярне технічне обслуговування, несправність обладнання, незаконне викрадення електроенергії, температура або раптові зміни навколишнього середовища. Формування моделі реакції виявлення ненормального навантаження для надання керівних думок щодо електропостачання з боку попиту є одним із напрямів майбутніх досліджень. Для прогнозування даних про потужне навантаження подальше розпаралелювання алгоритму прогнозування даних про потужне навантаження є ще одним ключовим напрямком наступної роботи. Будучи важливою частиною аналізу даних про електроживлення, дослідження виявлення аномалій електронавантаження та методів прогнозування в середовищі хмарних обчислень все ще мають величезні дослідницькі можливості, які чекають на вивчення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Lin, W.; Wu, G.; Wang, X.; Li, K. An artificial neural network approach to power consumption model construction for servers in cloud data centers. *IEEE Trans. Sustain. Comput.* 2019, 5, 329–340.
2. Liu, Y.; Liang, S.; He, C.; Zhou, Z.; Fang, W.; Li, Y.; Wang, Y. A Cloud-computing and big data based wide area monitoring of power grids strategy. *IOP Conf. Ser. Mater. Sci. Eng.* 2019, 677, 042055.
3. Raza, M.Q.; Khosravi, A. A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renew. Sustain. Energy Rev.* 2015, 50, 1352–1372.
4. Santos, M.A.G.; Munoz, R.; Olivares, R.; Filho, P.P.R.; Del Ser, J.; de Albuquerque, V.H.C. Online heart monitoring systems on the internet of health things environments: A survey, a reference model and an outlook. *Inf. Fusion* 2020, 53, 222–239.
5. Sharma, S.; Kotturu, P.K.; Narooka, P.C. Implication of IoT components and energy management monitoring. In *Swarm Intelligence Optimization: Algorithms and Applications*; JohnWiley & Sons, Inc.: New York, NY, USA, 2020; pp. 49–65.
6. Xu, Y.; Liu, C.C.; Schneider, K.P.; Tuffner, F.K.; Ton, D.T. Microgrids for service restoration to critical load in a resilient distribution system. *IEEE Trans. Smart Grid* 2018, 9, 426–437.
7. AL-Jumaili, A.H.A.; Al Mashhadany, Y.I.; Sulaiman, R.; Alyasseri, Z.A.A. A Conceptual and Systematics for Intelligent Power Management System-Based Cloud Computing: Prospects, and Challenges. *Appl. Sci.* 2021, 11, 9820.
8. Miu, W.; Zhang, Z.; Wang, X.; Hou, J.; Sun, Y. A Real-time Detection Framework for Abnormal Devices in the Power Internet of Things. *J. Phys.* 2022, 2166, 012057.
9. Jain, S.; Chandrasekaran, K. Industrial automation using internet of

things. In *Research Anthology on Cross-Disciplinary Designs and applications of Automation*; IGI Global: New York, NY, USA, 2022; pp. 355–383.

10. Zaheeruddin; Singh, K.; Amir, M. Intelligent Fuzzy TIDF-II Controller for Load Frequency Control in Hybrid Energy System. *IETE Tech. Rev.* 2021, 39, 1355–1371.

11. Manimegalai, C.T.; Kalimuthu, K.; Gauni, S. Toward integrating bidirectional multiband data and power transmission using double clad optical fibers for the next generation disaster resilient managing communication systems. *Microw. Opt. Technol. Lett.* 2022, 64, 816–820.

12. Jia, X. Research on network abnormal data flow mining based on improved cluster analysis. *Distrib. Parallel. Databases* 2021, 40, 797–813.

13. Ahamed, R.; Habeeb, A.; Nasaruddin, F.; Gani, A.; Abaker, I.; Hashem, T.; Ahmed, E.; Imran, M. International Journal of Information Management Real-time big data processing for anomaly detection: A Survey. *Int. J. Inf. Manag.* 2018, 45, 289–307.

14. Lin, Z.; Xiangping, L.; Wenzhong, C.; Haoyue, P. Computer aided analysis and control of power system based on data mining technology. In *Proceedings of the 2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers, Dalian, China, 14–16 April 2021*; pp. 1258–1261.

15. Malik, M.I. Cloud Computing-Technologies. *Int. J. Adv. Res. Comput. Sci.* 2018, 9, 379–384.

16. Li, X.; Liu, H.; Wang, W.; Zheng, Y.; Lv, H.; Lv, Z. Big data analysis of the internet of things in the digital twins of smart city based on deep learning. *Futur. Gener. Comput. Syst.* 2022, 128, 167–177.

17. Gharehpasha, S.; Masdari, M.; Jafarian, A. Power efficient virtual machine placement in cloud data centers with a discrete and chaotic hybrid optimization algorithm. *Cluster Comput.* 2021, 24, 1293–1315.

18. Li, Y.; Shi, F.; Zhang, H. Panoramic synchronous measurement system for wide-area power system based on the cloud computing. In *Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications*

(ICIEA), Wuhan, China, 31 May–2 June 2018; pp. 764–768.

19. Wang, Y.; Chen, S. An approach to smart grid online data mining based on cloud computing. *Int. J. Simul. Syst. Sci. Technol.* 2016, 17, 17.1–17.5.

20. Sun, C.; Wang, X.; Zheng, Y.; Zhang, F. A framework for dynamic prediction of reliability weaknesses in power transmission systems based on imbalanced data. *Int. J. Electr. Power Energy Syst.* 2020, 117, 105718.

21. Sliwa, J. Assessing complex evolving cyber-physical systems (case study: Smart medical devices). *Int. J. High Perform. Comput. Netw.* 2019, 13, 294.

22. Urooj, S.; Singh, T.; Amir, M.; Tariq, M. Optimal Design of Power Transformer with Advance Core Material using ANSYS Technique. *Eur. J. Electr. Eng. Comput. Sci.* 2020, 4, 1–17.

23. Bekemeier, B.; Park, S.; Whitman, G. Challenges and lessons learned in promoting adoption of standardized local public health service delivery data through the application of the Public Health Activities and Services Tracking model. *J. Am. Med. Inform. Assoc.* 2019, 26, 1660–1663.

24. Sagheer, N.S.; Yousif, S. Canopy with k -means Clustering Algorithm for Big Data Analytics. *AIP Conf. Proc.* 2021, 2334, 070006.

25. Tao, D.; Lin, Z.; Wang, B. Load feedback-based resource scheduling and dynamic migration-based data locality for virtual hadoop clusters in openstack-based clouds. *Tsinghua Sci. Technol.* 2017, 22, 149–159.

26. Dong, F.; Guo, X.; Zhou, P.; Shen, D. Task-aware flow scheduling with heterogeneous utility characteristics for data center networks. *Tsinghua Sci. Technol.* 2019, 24, 400–411.

27. Javed, A.; Larijani, H.; Ahmadiania, A.; Gibson, D. Smart Random Neural Network Controller for HVAC Using Cloud Computing Technology. *IEEE Trans. Ind. Inform.* 2017, 13, 351–360.

28. Eddoujaji, M.; Samadi, H.; Bohorma, M. Data Processing on Distributed Systems Storage Challenges. In *Smart Innovation, Systems and Technologies*; Springer: Berlin, Germany, 2022; Volume 237, pp. 795–811.

29. Yang, C.; Huang, Q.; Li, Z.; Liu, K.; Hu, F. Big Data and cloud

computing: Innovation opportunities and challenges. *Int. J. Digit. Earth* 2017, 10, 13–53.

30. Zhang, Y.; Lang, Y.; Yang, S.; Zhao, K.; Liu, P.; Han, F. Big Data Storage Technology suitable for the Operation and Maintenance of New Generation Power Grid Dispatching Control System Operation. *IOP Conf. Ser. Earth Environ. Sci.* 2019, 300, 042084.

31. Kulkarni, N.; Lalitha, S.V.N.L.; Deokar, S.A. Real time control and monitoring of grid power systems using cloud computing. *Int. J. Electr. Comput. Eng.* 2019, 9, 941.

32. Ahmed, M.H.; Tiun, S.; Omar, N.; Sani, N.S. Short Text Clustering Algorithms, Application and Challenges: A Survey. *Appl. Sci.* 2022, 13, 342.

33. Alarifi, A.; Dubey, K.; Amoon, M.; Altameem, T.; El-Samie, F.E.A.F.E.A.; Altameem, A.; Sharma, S.C.C.; Nasr, A.A.A. Energy-Efficient Hybrid Framework for Green Cloud Computing. *IEEE Access* 2020, 8, 115356–115369.

34. Ali, A.H.; Mohammad, O.K.J. Impacting of the E-Platforms on the 4.0th Industrial Educational Revolution. In *Proceedings of the Pervasive Health: Pervasive Computing Technologies for Healthcare*, Cairo, Egypt, 24–26 March 2019.

35. Al-Nuaymy, A.I.; Hamed, W.A.; Ali, A.H. The Role of Social Media in Enhancing the Learning Process of Iraq Students. *ARPN J. Eng. Appl. Sci.* 2019, 14, 7142–7153.

36. Li, X.; Zhuang, W.; Zhang, H. Short-term Power Load Forecasting Based on Gate Recurrent Unit Network and Cloud Computing Platform. *Pervasive Health Pervasive Comput. Technol. Healthc.* 2020, 29, 1–6.

37. Yuan, J. An Anomaly Data Mining Method for Mass Sensor Networks Using Improved PSO Algorithm Based on Spark Parallel Framework. *J. Grid Comput.* 2020, 18, 251–261.

38. Deng, C.; Liu, J.; Liu, Y.; Yu, Z. Cloud computing based high-performance platform in enabling scalable services in power system. In

Proceedings of the 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Changsha, China, 13–15 August 2016; pp. 2200–2203.

39. Hu, L.; Yang, C.; Meng, X.; Pang, P. Short-Term Power Load Forecasting Method Based on Cloud Computing and Intelligent Algorithms. In *Lecture Notes on Data Engineering and Communications Technologies*; Springer: Berlin, Germany, 2022; Volume 98, pp. 518–525.

40. Litchfield, A.T.; Althouse, J. A systematic review of cloud computing, big data and databases on the cloud. In *Proceedings of the Americas Conference on Information Systems*, Savannah, GA, USA, 7–9 August 2014.

41. Wan, L.; Zhang, G.; Li, H.; Li, C. A Novel Bearing Fault Diagnosis Method Using Spark-Based Parallel ACO-K-Means Clustering Algorithm. *IEEE Access* 2021, 9, 28753–28768.

42. Aziz, K.; Zaidouni, D.; Bellafkih, M. Real-time data analysis using Spark and Hadoop. In *Proceedings of the 2018 International Conference on Optimization and Applications, ICOA 2018*, Mohammedia, Morocco, 26–27 April 2018; pp. 1–6.

43. Zhu, T.; Xiao, S.; Zhang, Q.; Gu, Y.; Yi, P.; Li, Y. Emergent Technologies in Big Data Sensing: A Survey. *Int. J. Distrib. Sens. Networks* 2015, 2015.

44. Jiang, H.; Wang, K.; Wang, Y.; Gao, M.; Zhang, Y. Energy big data: A survey. *IEEE Access* 2016, 4, 3844–3861.

45. Sami, S.; Sael, N. Extract Five Categories CPIVW from the 9V's Characteristics of the Big Data. *Int. J. Adv. Comput. Sci. Appl.* 2016, 7, 254–258.

46. Jayaprakash, J.S.; Balasubramanian, K.; Sulaiman, R.; Hasan, M.K.; Parameshachari, B.D.; Iwendi, C. Cloud data encryption and authentication based on enhanced Merkle hash tree method. *Comput. Mater. Contin.* 2022, 72, 519–534.

47. Lin, C.-H.; Amir, M.; Tariq, M.; Shahvez, M.; Alamri, B.; Alahmadi, A.; Siddiqui, M.; Beig, A.R. Comprehensive Analysis of IPT v/s CPT for Wireless EV Charging and Effect of Capacitor Plate Shape and Foreign Particle on CPT.

Processes 2021, 9, 1619.

48. Jimeno, J.; Anduaga, J.; Oyarzabal, J.; De Muro, A.G. Architecture of a microgrid energy management system. *Eur. Trans. Electr. Power* 2011, 21, 1142–1158.

49. Nizeyimana, E. *Remote Sensing and GIS Integration*; McGraw-Hill Professional Publishing: New York, NY, USA, 2020.

50. Sharma, B.S.S.P.M.; Kamath, H.R.; Brahmaiah Rama, V.S. Modelling of cloud based online access system for solar charge controller. *Int. J. Eng. Technol.* 2018, 7, 58–61.

51. Kong, X.; Zheng, Y.; Ouyang, M.; Li, X.; Lu, L.; Li, J.; Zhang, Z. Signal synchronization for massive data storage in modular battery management system with controller area network. *Appl. Energy* 2017, 197, 52–62.

52. Kabalcı, E.; Kabalcı, Y.; Siano, P. Design and implementation of a smart metering infrastructure for low voltage microgrids. *Int. J. Electr. Power Energy Syst.* 2022, 134, 107375.

53. Amir, M.; Zaheeruddin. ANN Based Approach for the Estimation and Enhancement of Power Transfer Capability. In *Proceedings of the 2019 International Conference on Power Electronics, Control and Automation (ICPECA)*, New Delhi, India, 16–17 November 2019; pp. 1–6.

54. Волк М., Кипаренко Д.О. та інші. Розподілене комп'ютерне моделювання в системах хмарних обчислень. *Вісник Херсонського національного технічного університету*. №1. 2024. с. 211-217. Фахове видання.