НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

М.О. КІКТЄВ, В.В. КОВАЛЬ, О.М. РОМАЩУК

ОБРОБКА ІНФОРМАЦІЇ В КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ ТА РОБОТОТЕХНІЧНИХ СИСТЕМАХ

КОНСПЕКТ ЛЕКЦІЙ

Частина 2

Київ – 2025

УДК 681.51

«Обробка інформації в комп'ютерно-інтегрованих та робототехнічних системах»

Частина 2

[**Кіктєв М.О.,** доцент, канд. техн. наук, **Коваль В.В.,** професор, доктор технчних наук, **Ромащук О.М.,** доктор філософії] – К., 2025. – 150 с.

У лекційних матеріалах представлені методи обробки інформації в комп’ютерно-інтегрованих та робототехнічних системах автоматизації. Інформаційні технології (ІТ) займають важливе місце у всіх сферах життя та діяльності людини. Особливе місце в різноманітті ІТ займають автоматизовані системи обробки інформації, основне призначення яких - автоматизація діяльності, пов'язаної зі зберіганням, передачею та обробкою інформації. Оскільки інформація є у світі найважливішим ресурсом, то й такі системи грають визначальну роль у будь-якій сфері діяльності (технічні, бухгалтерські, банківські, складські, адміністративно-управлінські автоматизовані системи). Сучасні системи автоматизації аграрного спрямування спираються на використання локальних та глобальних мереж, обробку графічної, відео- та звукової інформації, технології мультимедіа, технологій та систем

штучного інтелекту. Без такого роду систем важко уявити сучасне підприємство, незалежно від розміру та напрями діяльності. Це багато в чому визначає існуючий постійний попит у всіх галузях економіки на фахівців у галузі проектування, створення та використання систем обробки інформації.

Лекційний матеріал призначений для аспірантів, які навчаються за спеціальністю G7 (174) «Автоматизація та комп'ютерно-інтегровані технології», а також може бути корисним докторантам та фахівцям в області автоматизації та інформаційних технологій.

ЗМІСТ

- 4.1. Завдання кластеризації в автоматизації: виявлення аномалій, класифікація станів обладнання.
- 4.2. Алгоритм k-середніх (k-means).
- 4.3. Ієрархічна кластеризація.
- 4.4. Застосування в технічному зорі: сегментація зображень, виявлення об'єктів.

Лекція 5: Системи зберігання інформації. Бази даних

- 5.1. Роль баз даних (БД) в інтегрованих системах.
- 5.2. Моделі даних: ієрархічна, мережна, реляційна.
- 5.3. Архітектура «озеро даних» (Data Lake) для промисловості.

Лекція 6: Вибірка та аналіз даних з баз

- 6.1. Мова SQL для роботи з реляційними БД:
    - Основні оператори: SELECT, FROM, WHERE.
    - Сортування (ORDER BY), групування (GROUP BY).
    - Агрегатні функції (COUNT, SUM, AVG, MIN, MAX).
- 6.2. З'єднання таблиць (JOIN) для отримання комплексної інформації.

Лекція 7: Методи обробки зображень у комп’ютерному зорі

7.1. Характеристики якості зображення.

7.2. Схема просторово-часової обробки кольорового зображення

7.3 Імовірнісний опис зображень

Лекція 8: Огляд інтелектуальних комп’ютерно-інтегрованих системи автоматизації і робототехніки

8.1. Вступ

8.2. Структура комп’ютерно-інтегрованої системи робототехніки

8.3. Роль теорії автоматичного керування в інтелектуальних роботах

8.4. Інтеграція ML у робототехнічні системи

8.5. Системи навігації та локалізації (SLAM)

8.6. Планування руху

8.7. Інтелектуальні системи контролю якості

8.8. Роботизоване відчуття (Multimodal Perception)

8.9. Цифрові двійники та симуляція

8.10. Архітектура промислової комп’ютерно-інтегрованої системи

8.11. Безпека робототехнічних систем

8.12. Тенденції розвитку

8.13. Практичні завдання до лекції

Лекція 9. Основи захисту інформації в комп’ютерно-інтегрованих та робототехнічних системах

Лекція 1. Інформація як основа систем автоматизації та робототехніки

Поняття «дані» та «інформація». Властивості інформації

Чим є інформація? На дане питання немає однозначної відповіді. Термін «інформація» є контекстним, тобто його сенс залежить лише від контексту, в якому вони вживаються. Будь-яка інформація має динамічний характер, тому що існує тільки в моменти взаємодії методів і даних. Весь інший час інформація перебуває в стані даних. Це призводить до того, що вона існує тільки в момент протікання інформаційного процесу. У момент споживання одні й ті ж самі дані можуть надавати різну інформацію, яка залежить від ступеня адекватності взаємодіючих з ними методів. Наприклад, для студента, який не володіє певною мовою, написаний на цій мові текст дає тільки ту інформацію, яку можна отримати методом спостереження (наявність незнайомих символів, загальна кількість символів, тощо). Використання ж більш адекватних для даного випадку методів дасть іншу інформацію. Дані є результатом реєстрації об’єктивно існуючих сигналів, які викликані змінами в полях або матеріальних тілах. Тому можна сказати, що дані є об’єктивними. На відміну від даних, методи є суб’єктивними. Методи також можна розділити на штучні та природні. В основі штучних методів лежать алгоритми – упорядковані послідовності команд – складені і підготовлені людьми (певними суб’єктами). В

основі ж природних методів лежать біологічні властивості суб'єктів інформаційного процесу.

Отже, можна сказати, що інформація виникає і існує в момент взаємодії об'єктивних даних і суб'єктивних методів. Характерною особливістю інформації є те, що на властивості інформації впливають властивості методів, які взаємодіють з даними в ході інформаційного процесу, і властивості даних, які складають її змістовну частину. Це відрізняє її від інших об'єктів природи і суспільства. Інформація має безліч властивостей. При цьому кожна наукова дисципліна розглядає ті властивості інформації, які є найбільш важливими в рамках відповідної області знань. Найбільш важливими (з точки зору інформатики) представляються наступні **властивості:** повнота, об'єктивність, адекватність, достовірність, доступність та актуальність інформації.

Повнота інформації. Повнота інформації часто характеризує її якість, а також визначає достатність наявних даних для створення на їх основі нових даних або для прийняття рішень. Чим більш повними є дані, тим більш широкий діапазон методів можна використовувати.

Об'єктивність і суб'єктивність інформації. Оскільки методи є суб'єктивними, можна сказати, що поняття об'єктивності є відносним. Прийнято вважати більш об'єктивною саме ту

інформацію, в яку методи вносять найменший елемент суб’єктивності.

Адекватність інформації. Під даним терміном найчастіше розуміється ступінь відповідності реальному об’єктивному стану речей. На основі неповних або недостовірних даних, а також через помилки в методології вимірювання (реєстрації) може утворюватися неадекватна інформація (навіть у випадку достовірних та повних даних).

Достовірність інформації. Дані виникають у момент вимірювання та реєстрації сигналів. Проте в більшості сигналів окрім корисної складової дуже часто можна знайти який рівень сторонніх сигналів, які не несуть корисної інформації в межах поставленої задачі. Такі елементи сигналів називаються інформаційним шумом. Тому основною задачею, яку необхідно вирішити для підвищення достовірності даних, є більш чітке виділення корисної складової серед інформаційного шуму. На сьогодні це можна зробити за допомогою великої кількості методів обробки даних (наприклад, фільтрації) або за допомогою використання новітніх методологій та засобів реєстрації сигналів (первинних перетворювачів).

Доступність інформації. Доступність являється мірою, яка характеризує можливість отримати певну інформацію. Доступність

даних та доступність адекватних методів їх інтерпретації – основні фактори, які впливають на доступність інформації. Відсутність одного з цих факторів призводить до однакового результату, а саме: інформація виявляється недоступною.

Актуальність інформації. Актуальність характеризує ступінь відповідності інформації в поточний момент часу дійсності. Найчастіше інформаційні процеси являються розтягнутими в часі, тому достовірна та адекватна, але дуже застаріла інформація може приводити до прийняття помилкових рішень.

Правильність відбору та формування інформації з метою адекватного відображення властивостей об'єкта характеризує її **репрезентативність.** У даному випадку важливе значення мають: о обґрунтованість відбору істотних зв'язків і ознак явища, яке розглядається; о правильність концепції, на базі якої було сформульовано вихідне поняття. До істотних погрішностей інформації нерідко призводить порушення її репрезентативності.

Змістовність інформації відображає семантичну ємність, рівну відношенню кількості семантичної інформації в повідомленні до обсягу оброблюваних даних.

Стійкість – це здатність інформації реагувати на зміни вихідних даних без порушення їх необхідної точності. Так само, як і

репрезентативність, стійкість інформації обумовлена правильністю вибору методики її відбору і формування.

Розрізняють наступні види інформації в залежності від її типу носія:

мовна (акустична);

документальна;

телекомунікаційна.

Мовна інформація виникає в ході процесу розмови, акустична – при роботі, наприклад, систем звуковідтворення та звукопідсилення. Звукові коливання в діапазоні частот (наближено) від 200 Гц до 5 кГц є носіями мовної інформації. Документальна інформація може надаватися в графічному або буквено-цифровому вигляді на папері, а також в електронному вигляді на магнітних та інших носіях. Телекомунікаційна інформація циркулює в технічних засобах обробки і зберігання інформації, а також в каналах зв’язку при її передачі. Носієм інформації при її обробці технічними засобами і передачі по провідним каналам зв’язку є електричний струм, а при передачі по радіо і оптичному каналах – електромагнітні хвилі. Джерело інформації може виробляти безперервний сигнал, у цьому випадку інформація називається аналоговою (безперервною), або перервний (роздільний, дискретний) сигнал – інформація

називається дискретною. Форму безперервних залежностей, які швидко змінюються в часі, мають, наприклад, сигнали, що передаються по радіо і телебаченню, а також використовуються при магнітному записі, а також усі процеси, які проходять у природі. На противагу цьому дискретні сигнали використовуються, наприклад, у телеграфії, комп'ютерній та обчислювальній техніці та мають переважно імпульсну форму. Порівнюючи аналогову і дискретну форми подання інформації, неважко помітити, що при використанні аналогової форми для створення обчислювальної машини потрібно менше число пристроїв (кожна величина репрезентується одним, а не декількома сигналами), але ці пристрої будуть складнішими (вони повинні розрізняти значно більше число станів сигналу). Інформація, що циркулює в суспільстві, вимагає спеціальних засобів і методів зберігання, використання та обробки. У процесі еволюції інформації сформувалися нові наукові дисципліни, наприклад, кібернетика, біоніка, робототехніка та інші, що мають на меті вивчення закономірностей інформаційних процесів. Існує три підходи до вимірювання інформації: 1) Невимірювана інформація в побуті, при якому вважається, що будь-яка інформація є новою. 2) Технічний або об'ємний підхід – інформація представляється як повідомлення в формі знаків або сигналів, які оброблюються, перетворюються і зберігаються за допомогою технічних пристроїв. В обчислювальній техніці застосовуються дві стандартні одиниці

вимірювання інформації: біт і байт. Оскільки комп’ютер призначений для обробки великих обсягів інформації, то додатково використовують похідні одиниці – кілобайт (кб), мегабайт (Мб), гігабайт (Гб), тощо. У математиці префікс «кіло» означає тисячу, а префікс «мега» – мільйон. Але в обчислювальній техніці здійснюється прив’язка до прийнятої двійкової системи кодування. У силу цього один кілобайт дорівнює не 1000 байтів, а 210, тобто 1024 байта. Аналогічно, 1 Мб = 210 кб = 1024 кб = 220 байтів = 1 048 576 байтів. 1 Гб = 210 Мб = 1024 Мб = 220 кб = 230 байтів = 1 073 741 824 байти. 3) Імовірнісний підхід визначає, що інформація виступає у ролі знятої невизначеності.

Збільшення кількості інформації, тобто отримання нової, означає збільшення знання, що, в свою чергу, призводить до зменшення незнання або інформаційної невизначеності. За одиницю кількості інформації найчастіше приймають вибір одного з двох рівноймовірних повідомлень, а саме: 1 або 0; так або ні. Ця одиниця також названа бітом.

Інформацію слід вважати особливим видом ресурсу, при цьому мається на увазі тлумачення "ресурсу" як запасу якихось знань матеріальних предметів або енергетичних, структурних або яких-небудь інших характеристик предмета. На відміну від ресурсів, пов'язаних з матеріальними предметами, інформаційні ресурси є

невичерпними і припускають істотно інші методи відтворення і оновлення, ніж матеріальні З цієї точки зору можна розглянути такі властивості інформації:

1. запам'ятовуваність;

2. передавання;

3. відтворюваність;

4. перетворювання;

5. стертість.

Запам'ятовуваність - одне з найважливіших властивостей. Запам'ятовувати інформацію будемо називати макроскопічної (маючи на увазі просторові масштаби запам'ятовуючому осередку і час запам'ятовування). Саме з макроскопічної інформацією ми маємо справу в реальній практиці.

Передання інформації за допомогою каналів зв'язку (у тому числі з перешкодами) добре досліджена в рамках теорії інформації К. Шеннона. У даному випадку мається на увазі дещо інший аспект - здатність інформації до копіювання, тобто до того, що вона може бути "запам'ятати" другий макроскопічної системою і при цьому залишиться тотожною самій собі. Очевидно, що кількість інформації не повинно зростати при копіюванні.

Відтворюваність інформації тісно пов'язана з її передавання і не є її незалежною базовим властивістю. Якщо передання означає, що не слід вважати істотними просторові відносини між частинами системи, між якими передається інформація, то відтворюваність характеризує невичерпність і невичерпною інформації, тобто що при копіюванні інформація залишається тотожною самій собі.

Фундаментальна властивість інформації - **перетворюванність.** Воно означає, що інформація може міняти спосіб і форму свого існування. Копійованого є різновид перетворення інформації, при якому її кількість не змінюється. У загальному випадку кількість інформації в процесах перетворення змінюється, але зростати не може. Властивість **стертості** інформації також не є незалежним. Воно пов'язане з таким перетворенням інформації (передачею), при якому її кількість зменшується і стає рівною нулю. Даних властивостей інформації недостатньо для формування її заходи, так як вони ставляться до фізичного рівня інформаційних процесів.

Електронна форма представлення даних – це спосіб фіксації даних, який дозволяє їх збереження, обробку, розповсюдження та представлення користувачеві за допомогою засобів обчислювальної техніки. Усі застосування визначення «електронні» можна узагальнити за такими ознаками, як:

- подання даних в цифровому вигляді (текст, звук, зображення статичне або те, що рухається);

- необхідність програмних та апаратних засобів для сприйняття людиною даних (тобто, комп'ютерного обладнання та програмного забезпечення);

- необхідність телекомунікаційних засобів для отримання або розповсюдження даних.

До електронної інформації як такої належать: дані, відомості, повідомлення, сигнали, програми, інформаційні системи, інформаційні технології та ін.

Інформація (Information)

- **Визначення:** Інформація — це дані, які були оброблені, структуровані та інтерпретовані для надання їм смислу, контексту та цінності. Це відповідь на запитання: "Що це означає?".
- **Аналогія:** Інформація — це готовий будинок, побудований з цеглин-даних за певним проектом.
- **Приклади в автоматизації (на основі даних вище):**
  - Температура в печі становить 24.7°C, що є нижче заданої технологічної норми (25.0°C).
  - Механізм досяг кінцевого положення.

- На конвеєрі виявлено дефектний виріб з координатами (x=150, y=75).

Дані – це відомості, отримані шляхом вимірювання, спостереження, логічних або арифметичних операцій і представлені у формі, придатній для постійного зберігання, передачі та автоматизованої обробки. Це тексти, таблиці, ілюстрації, відомості про факти, явища і т. ін., представлені у буквеноцифровій, числовій, текстовій формах, які зберігаються в комп'ютері, можуть пересилатися і піддаватися обробці. Дані, призначені для передачі, називаються повідомленням. Одним із способів перетворення даних в повідомлення є запис його на матеріальному носії. Процес такого запису називається кодуванням. Сигнал – повідомлення про те, що відбувається або може відбутися; матеріальний носій інформації; сукупність умовних знаків та засобів їхнього передавання й приймання.

Дані (Data)

- **Визначення:** Дані — це необроблені, несистематизовані факти, числа, символи або сигнали, що не несуть конкретного смислового навантаження самі по собі.
- **Аналогія:** Дані — це цегла та будматеріали. Самі по собі вони ще не є будинком.
- **Приклади в автоматизації:**

- Показник з датчика температури: 24.7.
- Сигнал "1" з дискретного входу (кінцевий вимикач увімкнено).
- Набір пікселів з камери технічного зору.

Ключова відмінність

Аспект	**Дані**	**Інформація**
Сутність	Сирі факти, "сигнал"	Оброблені, осмислені дані, "повідомлення"
Структура	Неструктуровані	Структуровані, класифіковані
Контекст	Відсутній	Є контекст і мета
Корисність	Безперечна	Стає корисною після обробки
Приклад	101101	Температура перевищила норму: 101.1°C

Перетворення **Даних** на **Інформацію** — це основне завдання будь-якої системи обробки інформації в автоматизації. Цей процес включає фільтрацію, усереднення, порівняння з нормою, агрегацію та інші види аналізу.

Властивості Інформації

Для ефективної роботи систем автоматизації інформація повинна володіти низкою ключових властивостей.

1. Актуальність (Timeliness/Relevance)

- **Що це?** Відповідність інформації поточному моменту часу.
- **Чому важливо?** Управлінські рішення, особливо в реальному часі (наприклад, аварійне відключення), мають базуватися на актуальних даних. Запізніла інформація марна.
- **Приклад:** Інформація про те, що тиск у казані перевищив норму 5 хвилин тому, є неактуальною для запобігання аварії. Потрібні дані в режимі реального часу.

2. Достовірність (Accuracy/Reliability)

- **Що це?** Відповідність інформації реальному стану об'єкта або процесу. Відсутність помилок та спотворень.
- **Чому важливо?** Недостовірна інформація призводить до неправильних управлінських рішень, браку продукції та аварій.
- **Приклад:** Несправний датчик температури показує 50°C замість реальних 100°C. Система управління не включить охолодження, що призведе до перегріву.

3. Повнота (Completeness)

- **Що це?** Наявність усіх необхідних даних для прийняття рішення.

- **Чому важливо?** Неповна картина може призвести до неправильних висновків.
- **Приклад:** Для оцінки ефективності роботи насоса недостатньо знати лише його тиск. Потрібні також дані про витрату рідини, температуру, споживану потужність.

4. Доступність (Accessibility)

- **Що це?** Можливість отримати інформацію в потрібний момент потрібному користувачу або системі.
- **Чому важливо?** Інформація, до якої неможливо отримати доступ, марна.
- **Приклад:** Оператор має мати можливість швидко переглянути тренд тиску за останню годину. Історичні дані мають бути доступні для інженера-технолога для аналізу.

5. Узгодженість (Consistency)

- **Що це?** Відсутність суперечностей у інформації, що надходить з різних джерел або в різний час.
- **Чому важливо?** Суперечливі дані ускладнюють аналіз та прийняття рішень.
- **Приклад:** Один лічильник показує витрату 10 л/хв, а другий на тій же лінії — 12 л/хв. Система не може коректно працювати з такою неузгодженою інформацією.

6. Цінність (Value)

- **Що це?** Корисність інформації для досягнення конкретної мети.
- **Чому важливо?** Збір та обробка інформації вимагають ресурсів. Вона повинна приносити вигоду.
- **Приклад:** Інформація про середньодобову температуру навколишнього середовища може бути цінною для системи клімат-контролю, але безцільною для системи контролю тиску в замкнутій системі.

7. Зрозумілість (Clarity)

- **Що це?** Інформація має бути представлена у формі, зрозумілій для того, хто її сприймає (людини або іншої системи).
- **Чому важливо?** Незрозуміла інформація не може бути використана.
- **Приклад:** Показник "Статус: 0x00A3" незрозумілий оператору, тоді як повідомлення "**Попередження: Низький рівень мастила в насосі Р-101**" — зрозуміле і корисне.

Носії інформації

Інформація завжди пов'язана з матеріальним носієм. Носієм інформації може бути:

- будь-який матеріальний предмет (папір, камінь і т.д.); хвилі різної природи: акустична (звук), електромагнітна (світло, радіохвиля) і т.д.;

- речовина в різному стані: концентрація молекул в рідкому розчині, температура і т.д.

Машинні носії інформації: перфострічки, перфокарти, магнітні стрічки, і т.д.

Інформаційне моделювання і формалізація

Людство в своїй діяльності (науковій, освітній, технологічній, художній та інших) постійно створює і використовує моделі навколишнього світу. Суворі правила побудови моделей сформулювати неможливо, однак людство накопичило багатий досвід моделювання різних об’єктів і процесів.

Термін «модель» в реальному житті має безліч значень.

Наприклад, модель – це:

- новий об’єкт, який відображає істотні з точки зору мети моделювання сторони досліджуваного об’єкта чи явища;

- відтворення предмета в зменшеному або збільшеному вигляді (макет);

- схема, зображення або опис якого-небудь явища або процесу в природі і суспільстві;

- спрощена подоба реального об’єкта;

- об’єкт-заступник, який в певних умовах може заміняти об’єкторигінал, відтворюючи основні його властивості і характеристики та має істотні переваги для дослідження або зручність використання;

- фізичний або інформаційний аналог об’єкта, функціонування якого за певними параметрами є подібним функціонуванню реального об’єкта;

- новий об’єкт (реальний, інформаційний або уявний), відмінний від вихідного, який володіє істотними для цілей моделювання властивостями і в рамках цих цілей повністю замінює вихідний об’єкт.

Все різноманіття моделей ділиться на три класи:

-матеріальні (натурні) моделі – зменшені або збільшені копії, які відтворюють зовнішній вигляд об’єкта, що моделюється, його структуру або поведінку (деякі реальні предмети – макети, муляжі, еталони, тощо);

- уявні моделі (геометрична точка, математичний маятник, ідеальний газ, нескінченність, тощо);

- інформаційні моделі – опис модельованого об’єкта на одній з мов кодування інформації (словесний опис, схеми, креслення, карти, малюнки, наукові формули, програми, тощо).

Наведемо класифікацію інформаційних моделей.

Інформаційна (абстрактна) модель – опис об’єкта на будь-якій мові. Абстрактність моделі проявляється в тому, що її компонентами є сигнали і знаки (вірніше, закладений в них сенс), а не фізичні тіла.

Дескриптивна (описова) модель – словесний опис об’єкта, виражений засобами тієї чи іншої мови.

Математична модель – сукупність записаних на мові математики співвідношень (формул, нерівностей, рівнянь, логічних співвідношень, тощо), що визначають характеристики стану об’єкта в залежності від його елементів, властивостей, параметрів, зовнішніх впливів; наближений опис об’єкта, виражений за допомогою математичної символіки.

Статичні моделі відображають об’єкт в якийсь момент часу без урахування змін, які відбуваються з ним, тобто відображають його таким, що знаходиться в стані спокою або рівноваги (відсутній параметр часу).

Динамічні моделі описують поведінку об’єкта в часі.

Детерміновані моделі відображають процеси, в яких відсутні випадкові величини.

Імовірнісні (стохастичні) моделі – опис об’єктів, поведінка яких визначається випадковими впливами (зовнішніми чи внутрішніми); опис імовірнісних процесів і подій, характер зміни яких у часі точно передбачити неможливо.

Імітаційна комп'ютерна модель – окрема програма, сукупність програм, програмний комплекс, що дозволяє за допомогою послідовності обчислень і графічного відображення їх результатів відтворювати (імітувати) процеси функціонування об'єкта, системи об'єктів за умови впливу на об'єкт різних факторів.

Імітаційна алгоритмічна модель – змістовний опис об'єкта в формі алгоритму, що відображає структуру і процеси функціонування об'єкта в часі, а також враховує вплив випадкових факторів.

Гносеологічна модель – опис об'єктивних законів природи.

Концептуальна модель описує виявлені причинно-наслідкові зв'язки і закономірності, притаманні досліджуваному об'єкту, і які є суттєвими в рамках певного дослідження.

Сенсуальні моделі – моделі почуттів, емоцій, або моделі, що впливають на почуття людини (музика, поезія, живопис, танець, тощо).

Модель-аналог – аналог об'єкта, який веде себе як реальний об'єкт, але не виглядає як такий.

Моделювання – це:

- дослідження об'єктів пізнання за їх моделями;

- заміна реального об'єкта його подібним аналогом;

- побудова моделей реально існуючих об'єктів (предметів, явищ, процесів, тощо).

Потреба в моделюванні виникає в таких сферах людської діяльності як: пізнання, практична діяльність, спілкування. Аспекти моделювання характеризуються властивостями:

- поведінка – зміна зовнішнього вигляду і структури з плином часу;

- структура – перелік елементів і зазначення відносин між ними;

- зовнішній вигляд – набір ознак.

Визначимо етапи моделювання:

1. Постановка цілей моделювання.

2. Аналіз об’єкта моделювання, виділення всіх його відомих властивостей.

3. Аналіз виділених властивостей і визначення істотних з них.

4. Вибір форми представлення моделі.

5. Формалізація.

6. Аналіз отриманої моделі на суперечливість.

7. Аналіз адекватності отриманої моделі об’єкту і мети моделювання.

Типи інформації в системах автоматизації.

Вихідний фізичний сигнал є неперервною функцією часу. Такі сигнали, визначені в усі моменти часу, називають аналоговими (analog). Послідовність чисел, що представляє сигнал при цифровій

обробці, є дискретним рядом (discrete series) і не може цілком відповідати аналоговому сигналові. Числа, що складають послідовність, є значеннями сигналу в окремі (дискретні) моменти часу і називаються відліками сигналу (samples). Як правило, відліки беруться через рівні проміжки часу T, названі періодом дискретизації (або інтервалом, кроком дискретизації - sample time). Величина, зворотна періодові дискретизації, називається частотою дискретизації (sampling frequency): $f_д = 1/T$. Відповідна їй кругова частота визначається як $\omega_д = 2\pi/T$. В загальному випадку представлення сигналу набором дискретних відліків приводить до втрати інформації, тому що ми нічого не знаємо про поведінку сигналу в проміжках між відліками. Проте при забезпеченні виконання умов теореми Котельнікова такої втрати інформації не відбувається й аналогові сигнали можуть бути точно відновлені за значеннями своїх дискретних відліків. 4 Процес перетворення аналогового сигналу в послідовність відліків називається дискретизацією (sampling), а результат такого перетворення — дискретним сигналом. При обробці сигналу в обчислювальних пристроях його відліки представляються у вигляді двійкових чисел, що мають обмежене число розрядів. В наслідок цього, відліки можуть приймати лише кінцеву множину значень і, отже, при представленні сигналу неминуче відбувається його округлення. Процес перетворення відліків сигналу в числа називається

квантуванням за рівнем (quantization), а помилки округлення, що при цьому виникають, – помилками (або шумами) квантування (quantization error, quantization noise). Сигнал, дискретний у часі, але не квантований за рівнем, називається дискретним (discrete-time) сигналом. Сигнал, дискретний у часі і квантованний за рівнем, називають цифровим (digital) сигналом. Сигнали, квантовані за рівнем, але неперервні в часі, на практиці зустрічаються рідко.

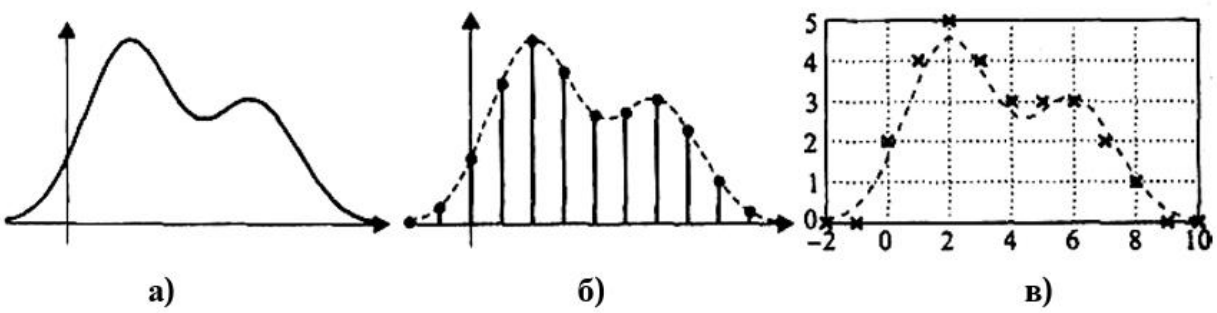


Рисунок 1.1 - Аналоговий (а), дискретний (б) і цифровий (в) сигнали

Розгялнемо такі типи інформації як, аналогова, дискретна, цифрова.

Аналогова інформація — це безперервна інформація, яка може набувати будь-якого значення в межах певного діапазону. Вона змінюється плавно в часі та просторі.

Ключова характеристика: Безперервність та нескінченна кількість можливих значень.

Аналогія: Рух стрілки аналогового годинника плавно відображає безперебний потік часу.

Джерела в системах автоматизації:

Температура (термопара, терморезистор)

Тиск (п'єзоелектричний датчик)

Рівень рідини або сипучого матеріалу

Витрата речовини

Освітленість (фоторезистор)

Швидкість обертання (тахогенератор)

Подання сигналу:

Найпоширеніший спосіб — за допомогою струму (4-20 мА) або напруги (0-10 В).

Приклад: Датчик температії з діапазоном 0-100°C формує струмовий сигнал 4-20 мА.

4 мА = 0°C

12 мА = 50°C

20 мА = 100°C

Переваги:

Можливість точно відобразити будь-який стан параметра.

Простота та низька вартість багатьох типів аналогових датчиків.

Недоліки:

Схильність до перешкод: Сигнал може спотворюватися електричними шумами на лінії передачі.

Обмежена точність: Точність залежить від якості датчика та АЦП.

Складність обробки: Для роботи з нею в мікропроцесорах потрібне попереднє перетворення.

Дискретна інформація — це цифрова інформація, яка може набувати лише обмежену кількість значень, найчастіше два: «Увімкнено»/«Вимкнено», «Так»/«Ні», «1»/«0». Її ще називають двійковою або сигналом двох станів.

Ключова характеристика: Два чітко визначені стани.

Аналогія: Вимикач світла — або увімкнено, або вимкнено.

Джерела в системах автоматизації:

Кінцеві вимикачі (об'єкт досяг положення)

Кнопки та перемикачі

Датчики наявності (індуктивні, ємнісні, оптичні)

Сигнали тривоги (аварія двигуна)

Реле та контактори

Подання сигналу:

Найчастіше подається як рівень напруги:

0 В (або близько 0 В) = Логічний "0" = "Вимкнено" = "False"

24 В (5В, 12В) = Логічний "1" = "Увімкнено" = "True"

Переваги:

Висока стійкість до перешкод: Системі важко сплутати 0В і 24В.

Простота обробки: Легко інтерпретується мікропроцесорами та ПЛК.

Надійність: Простота реалізації робить такі системи дуже надійними.

Недоліки:

Несе дуже обмежену інформацію (тільки факт стану, без його якісних характеристик).

Цифрова інформація (Digital Information)

Цифрова інформація — це інформація, представлена у вигляді послідовності дискретних кодів (зазвичай двійкових: 0 і 1). Вона є "оцифрованим" представленням аналогової величини або самостійним повідомленням.

Ключова характеристика: Квантування за рівнем та за часом. Представлення у вигляді коду.

Аналогія: Цифрові годинник, які показують час у вигляді чітких цифр (14:05:03), що стрибкоподібно змінюються.

Процес оцифрування:

Для перетворення аналогової інформації в цифрову використовується Аналого-Цифровий Перетворювач (АЦП):

Дискретизація за часом: Вимірювання миттєвого значення аналогового сигналу через рівні проміжки часу (частота дискретизації).

Квантування за рівнем: Привласнення кожному виміряному значенню найближчого рівня з обмеженої кількості.

Кодування: Перетворення квантованого рівня в двійковий код.

Наприклад, значення температури 27.4°C після оцифрування може бути представлене як двійковий код 11011 (що відповідає 27°C з певним кроком квантування).

Джерела в системах автоматизації:

Сучасні "розумні" датчики з цифровим виходом (наприклад, з інтерфейсом I²C, SPI).

Енкодери (кутові положення валу).

Системи технічного зору (зображення — це масив пікселів, кожен з яких має цифрове значення кольору).

Промислові мережі (Profibus, Ethernet/IP), що передають дані у вигляді пакетів.

Сканери штрих-кодів та RFID-термінали.

Переваги:

Найвища стійкість до перешкод: Дані можна передавати на великі відстані без втрат.

Легкість обробки, зберігання та копіювання: Ідеально підходить для комп'ютерів.

Висока точність: Залежить лише від розрядності АЦП.

Можливість стиснення та шифрування.

Недоліки:

Необхідність використання АЦП для роботи з аналоговими величинами.

Більша складність протоколів передачі даних.

Порівняльна таблиця

Характеристика	**Аналогова**	**Дискретна**	**Цифрова**
Характер змін	Безперервний	Стрибкоподібний	Стрибкоподібний
Можливі значення	Нескінченна кількість	2 стани	Обмежена кількість (2^n)
Приклад	Плавна зміна температури	Вимикатель	Текст, цифрове зображення
Стійкість до перешкод	Низька	Висока	Дуже висока

Характеристика	**Аналогова**	**Дискретна**	**Цифрова**
Обробка в ПЛК/ПК	Складна (потрібен АЦП)	Дуже проста	Пряма (ідеальна)
Типові сигнали	4-20 мА, 0-10 В	0/24 В	Послідовності 0 і 1 (пакети)

Розуміння типів інформації є критично важливим для проектування систем автоматизації:

Датчики вимірюють фізичні величини (переважно аналогові).

АЦП в ПЛК або в самому датчику перетворює аналоговий сигнал в цифровий код.

Дискретні сигнали використовуються для простих подій та станів.

Цифрові дані обробляються, аналізуються, передаються по мережах та зберігаються в базах даних для подальшого використання в системах керування та аналітики.

Контрольні запитання.

1. Що таке інформація?
2. Які є властивості у інформації?

3. Як класифікується інформація?

4. Які існують підходи до вимірювання інформації?

5. Що таке модель?

6. Як класифікуються моделі?

7. Які типи сигналів Ви знаєте?

Лекція 2: Збір інформації. Датчики та інтерфейси

Огляд первинних перетворювачів (датчиків).

Сучасна інформатизація суспільства вимагає побудови сучасних автоматизованих систем управління (АСУ) різними об’єктами. Стрімкий розвиток науки та техніки вимагає прогресивних рішень та пошуку нових моделей, методів, засобів та технологій побудови сучасних засобів автоматизації та комп’ютеризованих систем управління (КСУ). В умовах конкурентного ринку праці, сучасний фахівець з інформаційних технологій повинен володіти знаннями сучасного математичного апарату, методами моделювання та прогнозування, сучасними підходами до побудови автоматизованих систем управління, знати сучасні засоби автоматизації та володіти сучасними програмними засобами, що використовуються на всіх етапах життєвого циклу комп’ютерних систем. Широке використання АСУ обумовлене бажанням людства автоматизувати свою роботу з метою оптимізації та полегшення праці. На підприємствах з підвищеними ризиками небезпеки виключення участі людини у технологічних процесах є необхідністю, а не примхою. Сучасні підприємства та установи застосовують автоматизовані системи, як для управління підприємства, так, і для автоматизації технологічних процесів. Для автоматизації процесів проектування використовуються спеціальні

системи автоматизованого проектування (САПР). Основною тенденцією при цьому є використання та адаптація стандартних рішень з автоматизації з доопрацюванням і налаштуванням під конкретні потреби суб'єкта управління із врахуванням особливостей об'єкта управління. Основним та найбільш перспективним методом автоматизованого управління складними динамічними системами та процесами в життєво важливих і критичних з точки зору безпеки та надійності галузях (областях) є диспетчерське управління та збір даних (моніторинг та диспетчеризація) – SCADA(Supervisory Control And Data Acquisition) – системи. На принципах SCADA-систем будуються великі автоматизовані системи в промисловості, енергетиці, транспорті, в космічній та військових галузях, при автоматизації будівель та споруд.

Первинний перетворювач (датчик) — це пристрій, який перетворює вимірювану фізичну величину (температуру, тиск, переміщення тощо) у стандартний вихідний сигнал (електричний, пневматичний, цифровий), зручний для передачі, обробки та подальшого використання в системах контролю, керування та реєстрації.

Датчики температури, тиску, рівня, витрати

Це найпоширеніша група датчиків в промисловості, які формують основу систем автоматизації технологічних процесів.

Датчики температури

Принцип дії: Перетворення теплової енергії на електричний сигнал.

Основні типи:

Термопара: Два дроти з різних металів, з'єднані на кінці. При нагріванні місця з'єднання виникає ЕРС (термоелектричний ефект).

Переваги: Високі температури (до 2000°C), міцність, низька вартість.

Недоліки: Невисока точність, необхідність компенсації температури холодних кінців.

Термоопір (RTD - Resistance Temperature Detector): Використовує залежність електричного опору металу (зазвичай платини) від температури.

Переваги: Висока точність і стабільність, лінійна характеристика.

Недоліки: Вища вартість, менший діапазон температур, ніж у термопари.

Термістор: Напівпровідниковий резистор, опір якого різко змінюється з температурою.

Переваги: Висока чутливість, малі розміри.

Недоліки: Нелінійна характеристика, обмежений діапазон температур.

Пірометр (датчик випромінювання): Безконтактний датчик, що вимірює температуру за інфрачервоним випромінюванням об'єкта.

Переваги: Можливість вимірювання дуже високих температур і рухомих об'єктів.

Недоліки: Вимірює температуру поверхні, точність залежить від властивостей поверхні.

Датчики тиску

Принцип дії: Перетворення сили тиску на механічну деформацію, яка, у свою чергу, перетворюється на електричний сигнал.

Основні типи:

Тензорезистивні: Використовують мембрану, що деформується під тиском. На мембрані знаходяться тензорезистори, опір яких змінюється при деформації.

Переваги: Висока точність, надійність, широкий діапазон вимірювань.

Ємнісні: Вимірюють зміну ємності конденсатора, одна обкладка якого є рухомою мембраною.

Переваги: Висока чутливість, стабільність, стійкість до перевантажень.

П'єзоелектричні: Генерують електричний заряд пропорційно прикладеному тиску.

Переваги: Висока швидкодія, доброякісність для вимірювання динамічних тисків (наприклад, у двигунах).

Недоліки: Не підходять для вимірювання статичного тиску.

Датчики рівня

Принцип дії: Визначення положення межі поділу середовищ (наприклад, рідина-повітря).

Основні типи:

Поплавкові: Механічний поплавок, положення якого зміщується зі зміною рівня.

Ємнісні: Вимірюють зміну ємності між електродами при зміні діелектричної проникності середовища (наприклад, коли рідина замінює повітря).

Гідростатичні: Вимірюють тиск стовпа рідини на дні резервуара. По суті, це датчик тиску, відкалібрований у одиницях рівня.

Ультразвукові/Радарні: Вимірюють час проходження акустичного чи радіоімпульсу до поверхні рідини і назад.

Переваги: Безконтактність, універсальність.

Датчики витрати

Принцип дії: Вимірювання об'єму або маси середовища, що протікає через переріз трубопроводу за одиницю часу.

Основні типи:

Витратоміри змінного перепаду тиску (наприклад, стандартна діафрагма, сопло Вентурі): Вимірюють перепад тиску на місцевому опорі, який пропорційний квадрату витрати.

Турбінні: Потік обертає турбінку, частота обертання якої пропорційна витраті.

Вихреві: Вимірюють частоту утворення вихорів за тілом обтікання, яка пропорційна витраті.

Ультразвукові: Вимірюють різницю часу проходження ультразвукових імпульсів за течією і проти неї.

Коріолісові: Безпосередньо вимірюють масову витрату за рахунок ефекту Коріоліса (викривлення коливальної трубки).

Переваги: Висока точність, вимірювання масової витрати, незалежність від властивостей середовища.

Потенціометри, енкодери, тахогенератори

Ця група датчиків використовується для вимірювання механічних величин: положення (кутового/лінійного), швидкості та прискорення.

Потенціометри

Принцип дії: Резистивний подільник напруги, рухомий контакт (повзунок) якого переміщується вздовж резистивного елемента. Вихідна напруга пропорційна положенню повзунка.

Переваги: Простота, низька вартість, висока вихідна напруга.

Недоліки: Наявність механічного зносу, обмежений ресурс, чутливість до вібрацій.

Енкодери (обертові)

Принцип дії: Цифровий датчик кута повороту. Перетворює кутове положення валу в цифровий код.

Основні типи:

Інкрементальний (приріστовий): Генерує імпульси при обертанні. Підрахунком імпульсів визначають кут повороту, а фазу двох сигналів (А і В) — напрямок обертання.

Переваги: Висока роздільна здатність, простота.

Недоліки: Необхідність в "домашній" позиції (референтного імпульсу) для визначення абсолютного положення.

Абсолютний: Кожному кутовому положенню валу відповідає унікальний цифровий код (наприклад, двійковий або Грея).

Переваги: Миттєве визначення положення після включення живлення, стійкість до перешкод.

Тахогенератори

Принцип дії: Електрична машина, що працює в генераторному режимі. Вихідна напруга пропорційна швидкості обертання його валу.

Основні типи:

Постійного струму: Вихідна напруга постійного струму, полярність якої залежить від напрямку обертання.

Змінного струму (асинхронні): Вихідна напруга змінного струму, амплітуда і частота якої пропорційні швидкості обертання.

Переваги: Аналоговий вихід, висока лінійність, надійність.

Недоліки: Наявність пульсацій вихідної напруги (для ДС), обмежений діапазон швидкостей.

Сенсори технічного зору (2D, 3D)

Це складні електронно-оптичні системи, що імітують людський зір для отримання, аналізу та інтерпретації зображень.

2D-сенсори (двовимірні)

Принцип дії: Отримання плоскої (проекції) зображення об'єкта. Зазвичай використовуються ПЗЗ- або КМОН-матриці, аналогічні тим, що у цифрових фотоапаратах.

Основні застосування:

Контроль якості: Виявлення подряпин, дефектів, відсутніх елементів.

Ідентифікація: Зчитування штрих-кодів, QR-кодів, OCR (розпізнавання текстів).

Визначення положення (Positioning): Орієнтування деталей для роботів-маніпуляторів.

Вимірювання: Визначення лінійних розмірів, відстаней між об'єктами.

Переваги: Висока швидкість, хороша роздільна здатність, великий вибір обладнання.

Недоліки: Чутливість до освітлення, не дає інформації про глибину (об'єм).

3D-сенсори (тривимірні)

Принцип дії: Отримання мапи глибини (point cloud) — набору точок з координатами (X, Y, Z) у просторі.

Основні технології:

Стереобачення: Використовує дві камери для отримання бінокулярного зображення і розрахунку глибини за допомогою тріангуляції.

Структуроване підсвічування: Проєкція на об'єкт світлового шаблону (наприклад, сітки). Деформація цього шаблону на поверхні об'єкта аналізується камерою для розрахунку 3D-профілю.

Лідар (LiDAR): Вимірює відстань до об'єкта, визначаючи час повернення лазерного імпульсу. Скануючи променем, будує 3D-модель.

Time-of-Flight (ToF): Аналогічно лідару, але вимірює зсув фаз модульованого світлового променя, що дозволяє швидко отримувати повну мапу глибини за один кадр.

Основні застосування:

Об'ємні вимірювання: Визначення об'єму, форми.

Бін-пікінг: Ідентифікація та захоплення деталей з контейнера роботом.

Навігація роботів та безпілотників.

3D-сканування та реверс-інжиніринг.

Переваги: Наявність інформації про глибину, менша залежність від освітлення порівняно з 2D.

Недоліки: Вища вартість, більша обчислювальна складність.

Розгялнемо детальніше що ж таке LiDAR.

LiDAR (Light Detection and Ranging) — це технологія дистанційного зондування, яка використовує лазерні імпульси для вимірювання відстаней та створення детальних тривимірних карт об'єктів і навколишнього середовища. Вона працює шляхом випромінювання лазерного світла і вимірювання часу, необхідного для його повернення після відбиття від об'єкта, що дозволяє визначати відстань, форму та розмір. Технологія застосовується в автономних автомобілях, геодезії, картографії та мобільних пристроях для покращення функцій розширеної реальності (AR).

Як працює LiDAR

Випромінювання лазера: Система випромінює короткі лазерні імпульси в навколишнє середовище.

Відбиття: Імпульси відбиваються від об'єктів (землі, будівель, дерев тощо).

Вимірювання часу: Вбудований датчик вимірює час, який пройшов між випромінюванням імпульсу та його поверненням.

Обчислення відстані: На основі часу польоту фотона та швидкості світла, система обчислює відстань до об'єкта.

Створення моделі: Повторюючи цей процес тисячі разів, система створює точну тривимірну «хмару точок» (point cloud), яка представляє собою деталізовану модель навколишнього світу.

Сфери застосування

Автономні транспортні засоби: Виявлення перешкод, навігація та створення точних карт для безпечного водіння.

Геодезія та картографія: Створення високоточних топографічних карт, моніторинг змін рельєфу.

Археологія: Виявлення прихованих споруд під рослинністю.

Сільське господарство: Моніторинг посівів та стану ґрунту.

Архітектура та будівництво: Вимірювання та моделювання будівель.

Мобільні пристрої: Покращення роботи камери, функцій розширеної реальності (AR), точного фокусування в умовах низької освітленості.

2.2. Концепція «Інтернету речей» (IIoT) в промисловості.

Інтернет речей (англ. Internet of Things, IoT) — концепція мережі, що складається із взаємозв'язаних фізичних пристроїв, які мають вбудовані давачі, а також програмне забезпечення, що дозволяє здійснювати передачу і обмін даними між фізичним світом і комп'ютерними системами, за допомогою використання

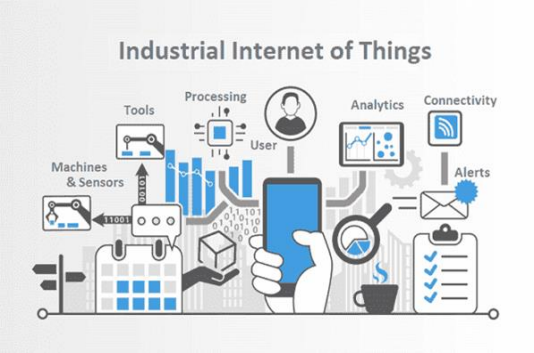
стандартних протоколів зв'язку. Окрім датчиків, мережа може мати виконавчі пристрої, вбудовані у фізичні об'єкти і пов'язані між собою через дротові чи бездротові мережі. Ці взаємопов'язані пристрої мають можливість зчитування та приведення в дію, функцію програмування та ідентифікації, а також дозволяють виключити необхідність участі людини, за рахунок використання інтелектуальних інтерфейсів.

Набуває поширення також термін англ. Internet of Everything, IoE — всеохопний, або всеосяжний інтернет. Це явище спричинило занепокоєння в конфіденційності інформації й сприяло появі нового терміну безпека інтернету речей.

Промисловий Інтернет речей (Industrial IoT, IIoT) - це один з найбільш великих сегментів Інтернету речей з точки зору кількості підключених пристроїв і ступеня корисності цих сервісів для виробництва і автоматизації підприємств. Цей сегмент традиційно служить операційно-технологічною базою. Сюди входять апаратні і програмні засоби моніторингу фізичних пристроїв. Традиційні завдання інформаційних технологій вирішуються інакше, ніж операційно-технологічні завдання. Операційні технології (OT) зосереджені на оцінці продуктивності, часу безвідмовної роботи,

зборі даних і відповідної реакції в режимі реального часу, а також безпеки систем. Інформаційні технології спрямовані на безпеку, групування, сервіси та надання даних. Оскільки Інтернет речей починає займати важливе місце в сфері виробництва і промисловості, світи IT і OT об'єднаються, особливо в області діагностичного обслуговування тисяч виробничих машин і верстатів, і зможуть забезпечувати безпрецедентним обсягом даних приватні та публічні хмарні інфраструктури.

До характеристик цього сегмента відноситься необхідність надавати операційно-технологічної системи готові рішення в режимі реального часу або майже в режимі реального часу. Це означає, що у всьому, що стосується виробничого цеху, головним параметром для Інтернету речей буде час відгуку. Крім того, важливу роль будуть грати тривалість простою і безпеку. Це має на увазі потребу в запасі потужності і, ймовірно, в наявності приватних хмарних мереж і сховищ даних. Промисловий Інтернет речей - це один з сегментів на цьому ринку що найбільш швидко розвивається. Важливою особливістю цього напрямку є те, що він спирається на старі технології, тобто на апаратні і програмні засоби, які не можна назвати актуальними. Часто 30-річні



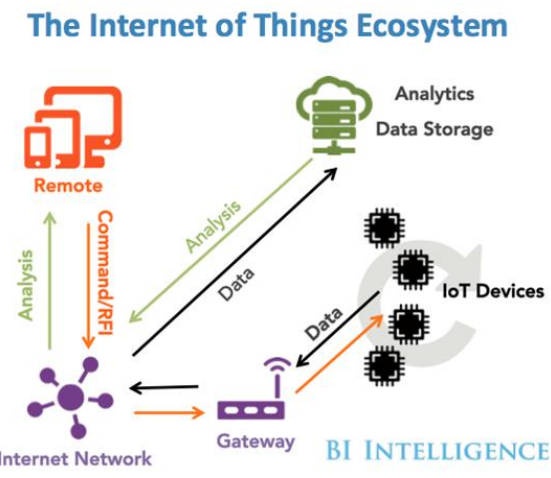
виробничі станки працюють на послідовних інтерфейсах RS485, а не на сучасній бездротовій комірчастій архітектурі.

Приклади застосування Промислового Інтернету Речей.

- профілактичне обслуговування промислового обладнання;
- зростання продуктивності завдяки попиту в реальному часі;
- енергозбереження;
- системи безпеки, такі як вимірювання температури, вимірювання тиску і контроль над витоком газу;
- експертна система для виробничого цеху.

Екосистема Інтернету речей

До екосистеми Інтернету речей відносяться усі засоби, сервіси і технології, які використовуються в Інтернеті речей.



До них можна віднести:

- sensors (розумні датчики/виконавчі механізми): вбудовані системи, операційні системи реального часу, джерела безперебійного живлення, мікро-електромеханічні системи (МЕМС);

- системи зв'язку з датчиками: зона охоплення бездротових персональних мереж становить від 0 см до 100 м. Для обміну даними між датчиками застосовуються низькошвидкісні малопотужні інформаційні канали, які часто побудовані не на протоколі IP;

- локальні обчислювальні мережі (LAN): зазвичай це системи обміну даними на основі протоколу IP, наприклад, 802.11 Wi-Fi-мережу для швидкої радіозв'язку, часто це пірингові або зіркоподібні мережі;

- агрегатори, маршрутизатори (routers), шлюзи (gateways), пограничні пристрої (Edge Device) : постачальники вбудованих систем, самі бюджетні складові (процесори, динамічна оперативна пам'ять і система зберігання даних), виробники модулів, виробники пасивних компонентів, виробники тонких клієнтів, виробники стільникових і бездротових радіосистем, постачальники міжплатформового програмного забезпечення, розробники інфраструктури туманних обчислень, інструментарій для граничної аналітики, безпеку граничних пристроїв, системи управління сертифікатами;

- глобальна обчислювальна мережа: оператори стільникового зв'язку, оператори супутникового зв'язку, оператори малопотужних глобальних мереж (Low- Power Wide-Area Network, LPWAN). Зазвичай застосовуються транспортні протоколи Інтернету для IoT і мережевих пристроїв (MQTT, CoAP і навіть HTTP);

- хмара: інфраструктура в якості постачальника послуг, платформа в якості постачальника послуг, розробники баз даних, постачальники послуг потокової і пакетної обробки даних, інструменти для аналізу даних, програмне забезпечення в якості постачальника послуг, постачальники озер даних, оператори програмно-визначених мереж / програмно-визначених периметрів, сервіси машинного навчання;

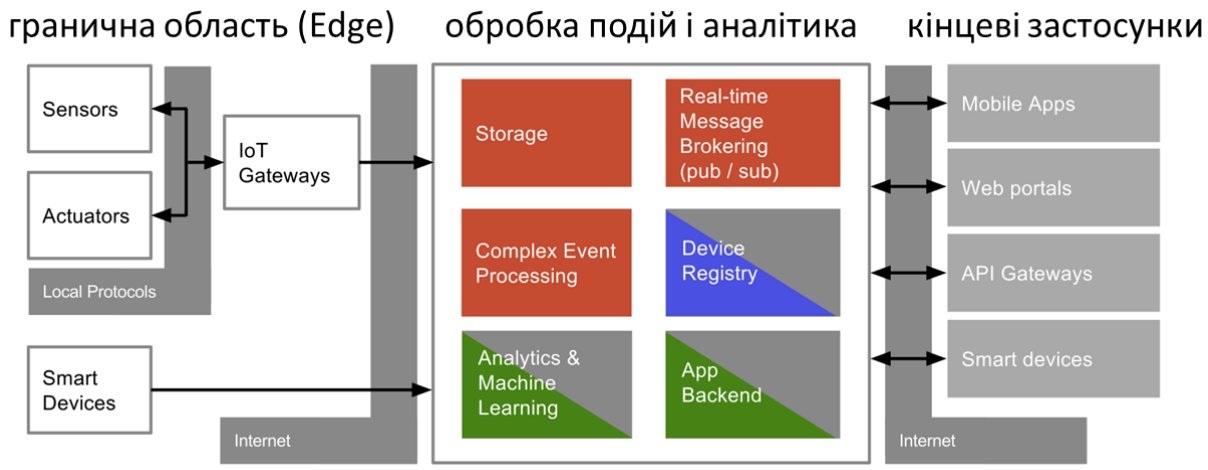
- сервіси аналізу даних: величезні масиви інформації передаються в хмару. Робота з великими обсягами даних і отримання з них користі - це завдання, що вимагає комплексної обробки подій, аналітики і прийомів машинного навчання;

- безпека (security): при зведенні всіх елементів архітектури воєдино постають питання кібербезпеки. Безпека стосується кожного компонента: від датчиків фізичних величин до ЦПУ і цифрового апаратного забезпечення, систем радіозв'язку і самих протоколів передачі даних. На кожному рівні необхідно забезпечити безпеку, достовірність і цілісність. У цьому ланцюзі не повинно бути слабких ланок, оскільки Інтернет речей стане головною мішенню для атак хакерів в світі.

Архітектура Інтернету Речей

Архітектура Інтернету речей відрізняється в залежності від реалізації. Тим не менше вона дещо схожа на архітектуру класичних систем АСУТП. Один із прикладів архітектури показаний на

рисунку.

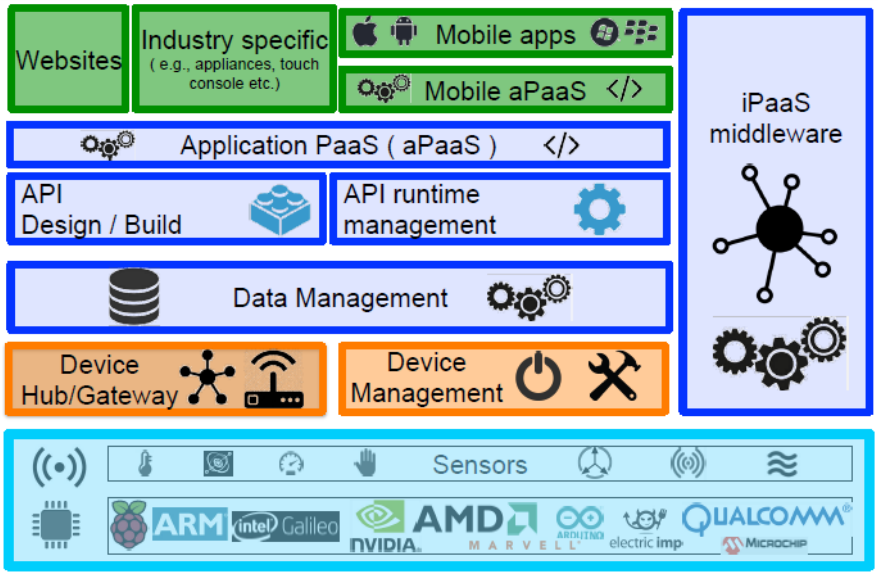


Взаємодія з «речами» відбувається через датчики (sensors) та виконавчі механізми (Actuators), аналогічно як це робиться в АСУТП для будь якого об’єкту керування. Ці датчики разом з усією інфраструктурою для інтеграції з рівнем обробки подій через мережу Internet формують так звану граничну область (Edge).

Події (дані) що поступають з граничної області зберігаються і обробляються відповідно до задачі (рівень обробки подій і аналітики, event processing, Platform). На цьому рівні події(дані) зберігаються (storage), обробляються (Event Processing), перенаправляються потрібним додаткам (Real-Time Message Brokering, Stream Processing). Додатково на цьому рівні відбувається адміністрування та керування пристроями з граничної області (Device Registry, Edge Device Management). Події (дані) обробляються з використанням аналітичних сервісів (Analytics) на основі них проводиться машинне навчання (Machine Learning), що

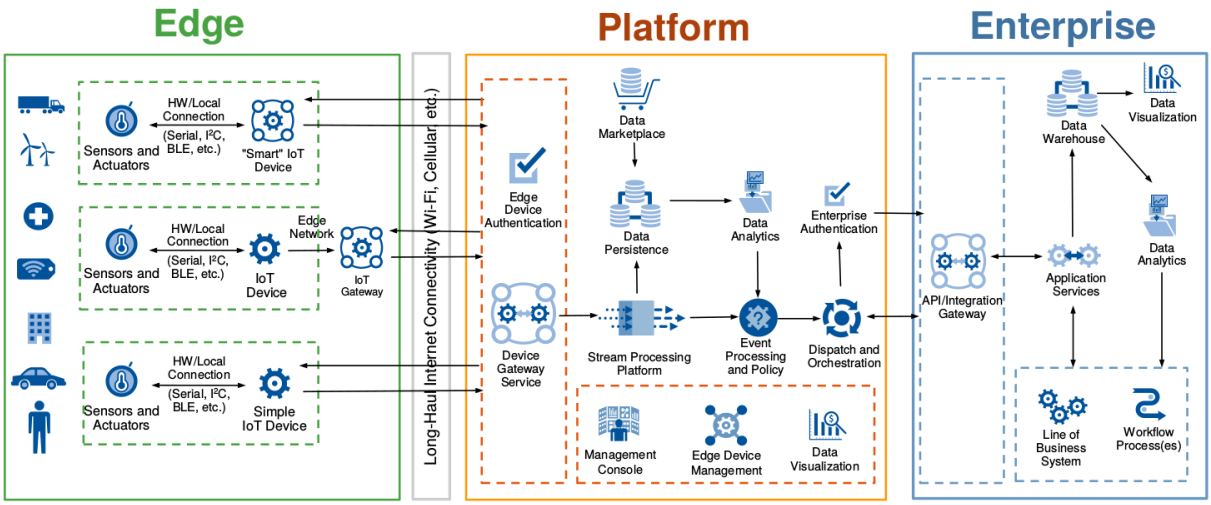
дозволяє зробити певні висновки про об’єкт. Цей рівень як правило реалізований з використанням хмарних (Cloud) або туманних (Fog) обчислень. Якщо провести аналогію с АСУТП, то це рівень контролерів та SCADA (за виключенням функцій HMI). Отримання результатів, контроль, віддалене керування та адміністрування системи проводиться через кінцеві застосунки з використанням Internet. Цей рівень можна умовно порівняти з HMI в АСУТП.

На рисунку показана подібна наведеній вище архітектура, однак у вигляді сервісів. На ньому область Edge представлений у вигляді датчиків (Sensors), Device Hub/Gateway (збір та маршрутизація даних) та Device Management (керування пристроями). Останні частково виконуються як хмарні обчислення так і на граничних пристроях. Усі функції збереження та первинної обробки подій (даних) зведені до Data Management. Усі інші функції обробки, в тому числі аналітичні показані як додатки PaaS, що взаємодіють з



сервісами керування даних через API (Application Program Interface).

Ще один приклад архітектури Інтернету Речей показаний на рис.3. Як видно, усі наведені архітектури мають спільні риси: наявність трьох рівнів, подібні функції, наявність хмарних обчислень, використання Інтернету як інтеграційного рівня.



Контрольні запитання

1. Який тип датчика температури ви б обрали для точного вимірювання в діапазоні від -50 до +200 °C і чому? Порівняйте з іншими типами.
2. Поясніть принцип роботи ємнісного датчика рівня. Як він відрізняється від гідростатичного?
3. Яка ключова відмінність між інкрементальним та абсолютним енкодером? Наведіть приклад застосування, де абсолютний енкодер є необхідним.

4. Чому для безконтактного вимірювання масової витрати суспензії (тверді частинки в рідині) краще підійде коріолісовий витратомір, а не ультразвуковий?
5. Яку перевагу має 3D-сенсор технічного зору з технологією Time-of-Flight (ToF) перед класичним 2D-сенсором при задачі бін-пікінгу (вибір деталей з контейнера роботом)?
6. Сфера застосування Інтернет речей (IoT)?
7. Архітектура Інтернет речей (IoT)

Лекція 3: Фільтрація та попередня обробка сигналів

3.1. Необхідність фільтрації. Види завад та шумів.

Методи обробки та фільтрації сигналів є важливими компонентами радіоелектроніки та приладобудування. Вони використовуються для вилучення корисної інформації з сигналів, видалення небажаних шумів і покращення співвідношення сигнал/шум. У цій роботі ми обговоримо принципи обробки та фільтрації сигналів, їх застосування та різні типи методів обробки та фільтрації сигналів.

Обробка сигналів передбачає маніпуляції з сигналами для вилучення корисної інформації або покращення їхньої якості. Основні принципи обробки сигналів включають дискретизацію, квантування та цифрову обробку сигналів. Дискретизація - це процес перетворення сигналу безперервного часу в сигнал дискретного часу шляхом вимірювання сигналу через регулярні проміжки часу. Квантування - це процес перетворення неперервного амплітудного сигналу в дискретний шляхом присвоєння кожному відліку найближчого рівня квантування. Цифрова обробка сигналів передбачає маніпуляції з цифровими сигналами за допомогою алгоритмів для вилучення корисної інформації або покращення їхньої якості.

Фільтри використовуються для видалення небажаного шуму з сигналів та покращення їх якості. Основні принципи фільтрації включають аналіз частотної області, передатні функції та проектування фільтрів.

Частотний аналіз передбачає представлення сигналів у частотній області за допомогою аналізу Фур'є. Передавальні функції використовуються для опису зв'язку між вхідним і вихідним сигналами фільтра. Проектування фільтрів передбачає вибір відповідного типу фільтра і оптимізацію його параметрів для досягнення бажаних характеристик. 19Методи обробки та фільтрації сигналів використовуються в широкому спектрі застосувань, включаючи системи бездротового зв'язку, обробку аудіо та відео, системи медичної візуалізації та системи управління. У системах бездротового зв'язку методи обробки і фільтрації сигналів використовуються для вилучення корисної інформації з сигналів і усунення перешкод. В обробці аудіо- та відеосигналів ці методи використовуються для покращення якості сигналу та видалення шумів. У системах медичної візуалізації методи обробки та фільтрації сигналів використовуються для покращення якості зображень та зменшення шуму. У системах керування ці методи використовуються для фільтрації небажаних сигналів і підвищення продуктивності системи. Існує кілька типів методів обробки та фільтрації сигналів, включаючи аналогову та цифрову фільтрацію,

фільтрацію в часовій та частотній області, а також адаптивну фільтрацію. Аналогові фільтри розроблені з використанням пасивних компонентів, таких як резистори, конденсатори та котушки індуктивності. Цифрові фільтри розроблені з використанням алгоритмів цифрової обробки сигналів. Часові фільтри працюють з представленням сигналу в часовій області, тоді як частотні фільтри працюють з представленням сигналу в частотній області. Адаптивні фільтри призначені для підстроювання своїх параметрів на основі характеристик вхідного сигналу.

Отже, методи обробки та фільтрації сигналів відіграють життєво важливу роль у роботі радіоелектроніки та контрольно-вимірювальних систем. Використовуючи ці методи, інженери та науковці можуть витягувати корисну інформацію з сигналів, видаляти небажані шуми та покращувати якість сигналу. Різні типи обробки та фільтрації сигналів надають інженерам та науковцям широкий спектр можливостей для оптимізації роботи їхніх систем. З розвитком технологій методи обробки та фільтрації сигналів ставатимуть все більш важливими у проектуванні та розробці радіоелектроніки та контрольно-вимірювальних систем.

Завади та шуми класифікують за різними ознаками: за походженням, характером впливу та спектральними характеристиками.

1. Класифікація за джерелом походження

- **Внутрішні шуми:** Генеруються всередині самого електронного обладнання.
  - **Тепловий шум (шум Джонсона-Найквіста):** Виникає через тепловий рух електронів у резисторах та провідниках. Є фундаментальним і присутній у всіх пристроях. Його потужність пропорційна температурі та смузі частот.
  - **Дробовий шум (шум Шоткі):** Виникає через дискретну природу електричного заряду (електронів). Особливо помітний в напівпровідникових приладах (транзисторах, діодах).
  - **Шум 1/f (флікер-шум або "рожевий" шум):** Потужність шуму зменшується зі збільшенням частоти (~1/f). Переважає на низьких частотах і є основним джерелом шуму в постійному струмі (DC) та низькочастотних вимірюваннях.
- **Зовнішні завади:** Надходять з навколишнього середовища.
  - **Електромагнітні завади (EMI):** Випромінювання від радіопередавачів, мобільних телефонів, Wi-Fi-роутерів, блискавки, імпульсні перешкоди від електродвигунів, систем запалювання.
  - **Наведення:**
    - **Ємнісне наведення:** Через паразитну ємність між двома провідниками (наприклад, між силовим кабелем і сигнальним).

- **Індуктивне наведення:** Виникає через електромагнітну індукцію, коли змінний магнітний потік від одного провідника пронизує контур іншого.
  - **Заземлення (петлі заземлення):** Якщо різні частини системи мають різний потенціал "землі", то струм може текти по земляному проводу, створюючи заваду в сигнальних ланцюгах.
  - **Вібрація та механічні удари:** Можуть впливати на чутливі датчики (наприклад, п'єзоелектричні) або викликати мікрофонний ефект в кабелях.

2. Класифікація за характером впливу

- **Адитивні завади:** Просто додаються до корисного сигналу. Більшість шумів (тепловий, дробовий) є адитивними.

  Сигнал_з_завадою(t) = Корисний_сигнал(t) + Завада(t)

- **Мультиплікативні завади:** Впливають на параметри самого сигналу (наприклад, амплітуду чи фазу). Боротьба з ними складніша.

  Сигнал_з_завадою(t) = Корисний_сигнал(t) * Завада(t)

3. Класифікація за спектральними характеристиками

- **Узкосмугові (тональні) завади:** Мають визначену частоту або дуже вузький спектр. Наприклад, завада від мережі електроживлення 50 Гц (60 Гц).

- **Широкосмугові завади:** Займають широкий діапазон частот. Наприклад, імпульсні перешкоди від колекторного двигуна, тепловий шум.
- **Імпульсні завади:** Короткі за часом, але з великою амплітудою викиди. Наприклад, перешкоди від комутації реле, блискавки, електростатичні розряди.

Для боротьби з різними видами завад використовують різні типи фільтрів:

- **Фільтр низьких частот (ФНЧ):** Пропускає сигнали з частотою **нижче** певної частоти зрізу. Використовується для **придушення високочастотних шумів** (RF-завади, імпульсні перешкоди) і підготовки сигналу для АЦП.
- **Фільтр високих частот (ФВЧ):** Пропускає сигнали з частотою **вище** частоти зрізу. Корисний для **блокади постійної складової** або низькочастотного дрейфу, залишаючи швидкі зміни сигналу.
- **Смуговий фільтр:** Пропускає сигнали в певному частотному діапазоні. Ідеально підходить для **виділення корисного сигналу** відомої частоти (наприклад, з вимірювального моста) та придушення завад поза його смугою.
- **Загороджувальний фільтр (Reject Filter):** Пригнічує сигнали в дуже вузькому діапазоні частот. Класичний приклад — **фільтр-пробка на 50 Гц** для боротьби з мережевою завадою.

Цифрова фільтрація в реальному часі.

Терміном цифровий фільтр називають апаратну або програмну реалізацію математичного алгоритму, входом якого є цифровий сигнал, а виходом – інший цифровий сигнал, форма якого і/або амплітудна та фазова характеристики спеціальним чином модифіковані.

В аналогових системах під фільтром розуміють деякий лінійний пристрій зі специальною частотною характеристикою $\dot{K}(j\omega)$, яке перетворює вхідний сигнал $s(t)$ у вихідний $y(t)$(рис. 3.1), придушуючи або, навпаки, підсилюючи при цьому певні частоти в спектрі вхідного сигналу. Вихідний сигнал $y(t)$знаходиться як згортка вхідного сигналу $s(t)$і імпульсної характеристики фільтра $h(t)$:

$$y(t) = s(t) * h(t) = \int_{-\infty}^{\infty} s(\tau)h(t-\tau)d\tau = \int_{-\infty}^{\infty} h(\tau)s(t-\tau)d\tau \quad s(t)$$

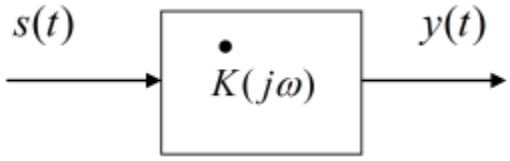


Рисунок 3.1 – Цифровий фільтр

За аналогією з аналоговим фільтром, цифровий фільтр (ЦФ, digital filter) перетворює послідовність відліків вхідного сигналу $\{s_k\}$ у числову послідовність вихідного сигналу $\{y_k\}$. Для ЦФ також вводять поняття імпульсної характеристики $\{h_k\}$, що є реакцією ЦФ на «одиничний імпульс (скачок)» тобто

$$(1,0,0,\ldots) \xrightarrow{ЦФ} (h_0, h_1, h_2, \ldots)$$

Імпульсну характеристику (pulse response characteristic) $\{h_k\}$ ЦФ можна трактувати як результат дискретизації безперервної імпульсної характеристики $h(t)$ відповідного аналогового фільтра-прототипу (рис. 3.2).

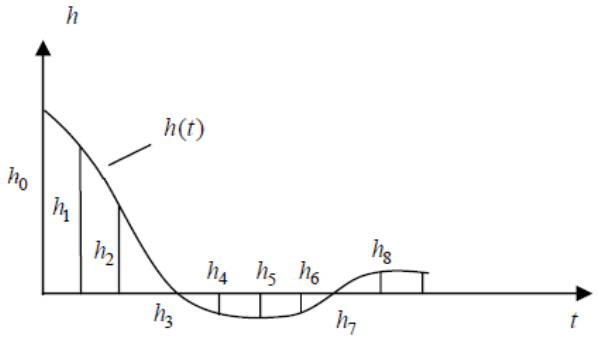


Рисунок 3.2 – Дискретизація імпульсної характеристики

Якщо взяти кінцеве число відліків $h(t)$, тоді отримаємо ЦФ із *кінцево-імпульсною характеристикою* (КІХ-фільтр, finite impulse response filtering). Якщо взяти нескінченне число відліків $h(t)$, отримаємо ЦФ із *безкінечно-імпульсною характеристикою* (БІХ-фільтр, infinite impulse response filtering).

Під фільтром зазвичай розуміють систему, що одні частоти пропускає, а інші затримує. Однак у техніці цифрової обробки сигналів поняття фільтра трактується більш широко. *Дискретним фільтром* називають довільну систему обробки дискретного сигналу, що має властивості лінійності й стаціонарності. Існують також фільтри зі змінними параметрами, наприклад, *адаптивні фільтри*, що змінюють свої параметри залежно від статистичних властивостей вхідного сигналу.

У загальному випадку, фільтр змінює в спектрі сигналу і амплітуди гармонік, і їх фази. Однак фільтри можна проектувати так, щоб вони не змінювали фазу сигналу. Такі фільтри називаються *фільтрами з лінійною фазою*. Це означає, що якщо вони і змінюють фазу сигналу, то роблять це так, що всі гармоніки сигналу зсуваються за часом на одну й ту ж величину. Таким чином, фільтри з лінійною фазою не спотворюють фазу сигналу, а лише зсувають весь сигнал в часі. Ядро згортки такого фільтра симетричне щодо своєї центральної точки.

Основна властивість будь-якого фільтру – це його частотна і фазова характеристики. Вони показують, як фільтр впливає на амплітуду і фазу різних гармонік оброблюваного сигналу. Якщо фільтр має лінійну фазу, то розглядається лише частотна характеристика фільтру. Зазвичай частотна характеристика зображується у вигляді графіка залежності амплітуди від частоти (в децибелах). Наприклад, якщо фільтр пропускає всі сигнали в смузі 0 ... 10 кГц без зміни, а всі сигнали в смузі вище 10 кГц подавляє в 2 рази (на 6 дБ), то частотна характеристика буде мати такий вигляд:

$$A(f) = \begin{cases} 0дБ, f < 10кГц \\ -6дБ, f > 10кГц \end{cases}.$$

Частотна характеристика в 0 дБ показує, що дані частоти фільтр пропускає без зміни. Ті частоти, амплітуда яких послаблюється фільтром в 2 рази, повинні мати амплітуду на 6 дБ менше. Тому їх

амплітуда становить -6 дБ. Якщо фільтр посилює частоти, то його частотна характеристика на цих частотах є позитивною.

Вихідний сигнал *y(k)* фільтра, що має нетривіальну частотну характеристикою, залежить від декількох відліків вхідного сигналу *x(k)*. У загальному випадку при обчисленні вихідного відліку використовується також деяка кількість попередніх відліків вихідного сигналу. Для фільтрів, що не використовують вихідні відліки, рівняння фільтрації має вигляд

$$y(k) = \sum_{i=0}^{m} b_i x_{(k-i)}$$

Такі фільтри називаються *не рекурсивними* (nonrecursive filter) або трансверсальними. Кількість відліків *m* називається порядком фільтра. Структурна схема нерекурсивного фільтра показана на рис.3.3. Імпульсна характеристика нерекурсивного фільтра визначається його коефіцієнтами $h(k) = b_k$. Так як в реальному пристрої кількість ліній затримки обмежено, а отже, і кількість коефіцієнтів, нерекурсивні фільтри відносять до класу КІХ-фільтрів.

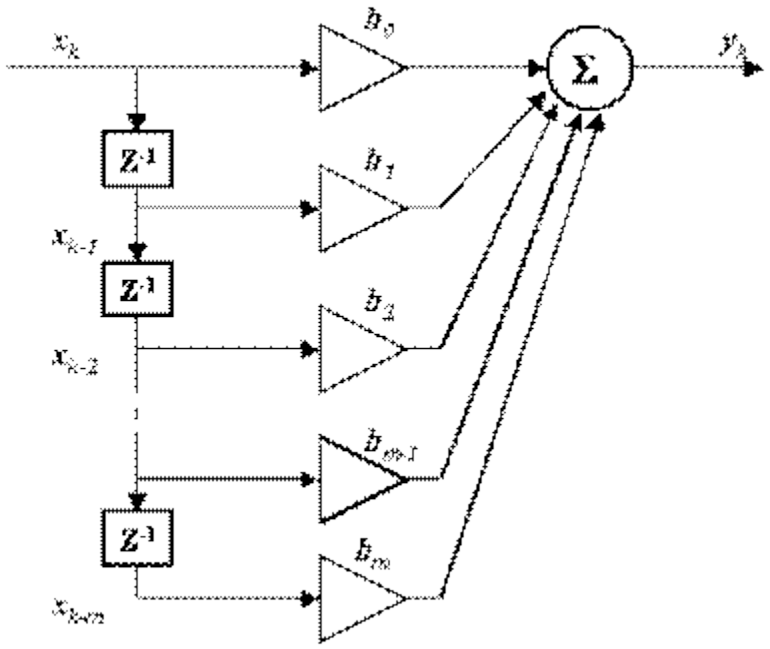


Рисунок 3.3 – Нерекурсивний фільтр

Фільтр, у якому також використовуються і вихідні відліки, називається *рекурсивним* (recursive filter) (рис.3.4). Рівняння рекурсивного фільтра має вигляд

$$y(k) = \sum_{i=0}^{m} b_i x_{(k-i)} - \sum_{i=0}^{n} a_i y_{(k-i)}$$

Наявність у схемі рекурсивного фільтра зворотних зв'язків дозволяє одержати безкінченну імпульсну характеристику, тому такі фільтри належать до класу БІХ-фільтрів. Проте такі фільтри при деяких умовах можуть бути нестійкими.

Під проектуванням (синтезом) цифрового фільтра розуміють вибір таких наборів його коефіцієнтів $\{a_i\}$ і $\{b_i\}$, які задовольняють заданим вимогам. У завдання проектування входить також і вибір потрібної структури фільтра з урахуванням необхідної точності обчислень.

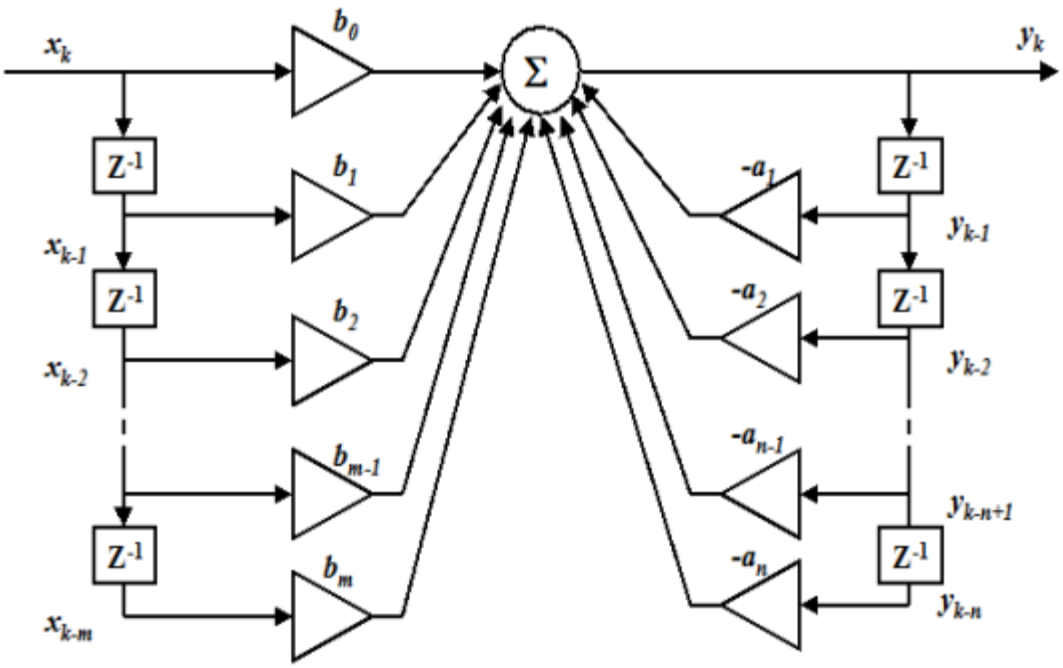


Рисунок 3.4 – Рекурсивний фільтр

У проектуванні та реалізації цифрових фільтрів застосовується безліч різноманітних підходів і методів, вибір яких залежить від багатьох факторів, зокрема – від того, як формулюється завдання фільтрації.

Найчастіше цифрова фільтрація застосовується для виділення сигналу або для відновлення сигналу. Виділення "корисного" сигналу необхідно, коли сигнал, що надходить у систему із зовнішнього середовища, змішаний із шумами, викликаними різноманітними фізичними процесами, що мають, як правило, випадковий характер. Відновлення сигналу необхідно через можливі спотворення сигналу, викликані роботою апаратури.

У зв'язку з тим, що теорія апроксимації ідеальних АЧХ аналоговими засобами добре розвинена, широке поширення одержали методи синтезу цифрових фільтрів по аналогових прототипах.

У прямих методах синтезу без використання аналогових прототипів часто використовується той факт, що ДПФ можна трактувати як обробку сигналу фільтром з відповідною імпульсною характеристикою.

Оскільки ДПФ лінійної згортки дорівнює добутку ДПФ послідовностей, що звертаються, алгоритм фільтрації в частотній області полягає в наступному:

- Послідовність відліків вхідного сигналу та імпульсна характеристика фільтра доповнюються нулями так, щоб довжини послідовностей стали рівними і не меншими, ніж сума довжин вихідних послідовностей мінус одиниця.
- Обчислюються ДПФ доповнених нулями послідовностей.
- Обчислені ДПФ поелементно перемножуються
- Обчислюється обернене ДПФ від результату перемножування.

Для підвищення ефективності фільтрація здійснюється в частотній області з використанням ШПФ.

При проектуванні цифрових фільтрів один із ключових моментів пов'язаний з особливостями технічної реалізації і рядом обмежень, обумовлених розрядністю цифрових обчислювальних пристроїв:

- шум квантування, що виникає при аналого-цифровому перетворенні;
- спотворення характеристик, що відбуваються при квантуванні коефіцієнтів цифрових фільтрів;
- переповнення розрядної сітки в процесі обчислень;
- округлення проміжних результатів обчислень.

Тому при проектуванні цифрових фільтрів важливо правильно вибрати способи й формати подання чисел (дійсні або комплексні числа, з фіксованою або плаваючою крапкою тощо), динамічний діапазон подання даних, обумовлений розрядністю регістрів устаткування, а також оцінити можливий вплив шумів і спотворень.

Фільтр нижніх частот

Фільтр нижніх частот (ФНЧ) є схемою, яка без змін передає сигнали нижніх частот, а на верхніх частотах забезпечує затухання сигналів і запізнення їх за фазою відносно вхідних сигналів. На рис. 1.1.1 наведена схема простого RC-фільтра нижніх частот.

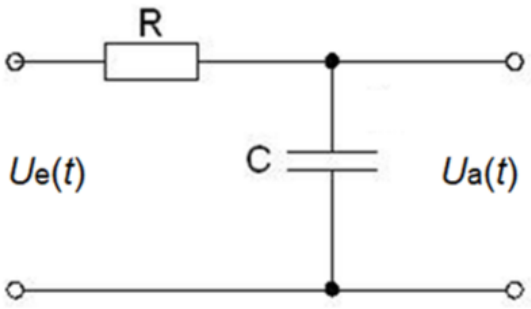


Рисунок 1.1.1 – Простий фільтр нижніх частот (ФНЧ)

Для розрахунку частотної характеристики схеми використаємо формулу відношення напруг, представлених в комплексній формі:

$$\underline{A}(j\omega) = \frac{\underline{U_a}}{\underline{U_e}} = \frac{1/(j\omega C)}{R+1/(j\omega RC)} = \frac{1}{1+j\omega RC}$$

З виразу (1.1.1), враховуючи, що $\underline{A} = |\underline{A}|e^{j\varphi}$ отримаємо

$$|\underline{A}| = \frac{1}{\sqrt{1+\omega^2R^2C^2}};\ \varphi = arctg(\omega RC)$$

Залежності, що описуються формулою (1.1.2) представлені на рис. 1.1.2.

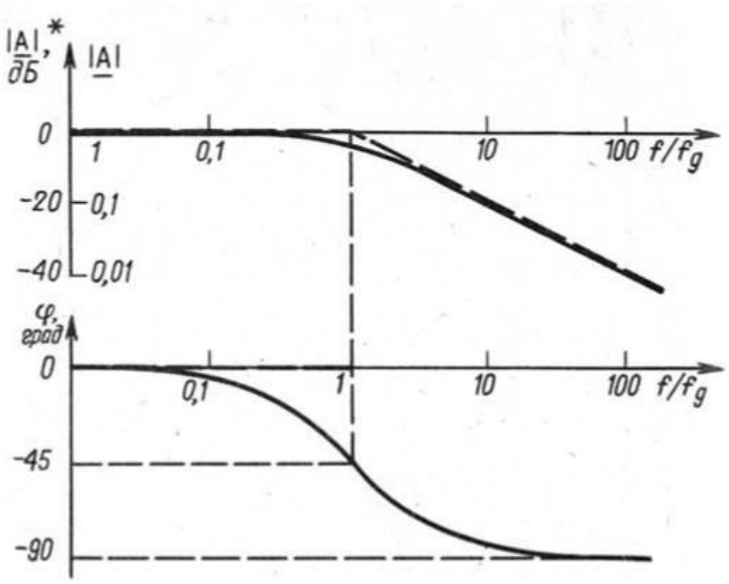


Рисунок 1.1.2 – Діаграма Боде для фільтру нижніх частот (ФНЧ)

Підставивши $|\underline{A}| = \frac{1}{\sqrt{2}}$, отримаємо вираз для визначення частоти зрізу

$$f_g = \omega_g \frac{1}{2\pi} = \frac{1}{2\pi RC}.$$

Фазовий зсув j на такій частоті, відповідно до (1.1.2), складає 45°.

Як видно з рис. 1.1.2, амплітудно-частотну характеристику $|\underline{A}| = \widehat{U_a}/\widehat{U_e}$, найпростіше скласти з двох асимптот:

1) $|\underline{A}| = 1 \mathrel{\widehat{=}} 0 дБ$, на нижніх частотах f << fg;

2) на високих частотах f>>fg, відповідно до формули (1.1.2), $|A| \approx (\omega RC)$, тобто коефіцієнт підсилення обернено пропорційний частоті.

При збільшенні частоти в 10 раз коефіцієнт підсилення зменшується в 10 раз (зменшується на 20 дБ на декаду або на 6 дБ на октаву);

3) $|\underline{A}| = 1/\sqrt{2} \mathrel{\widehat{=}} -3 дБ$ при $f = f_g$.

Для аналізу схеми в часовій області на вхід схеми (рис. 1.1.1) подається імпульс напруги (рис 1.1.3).

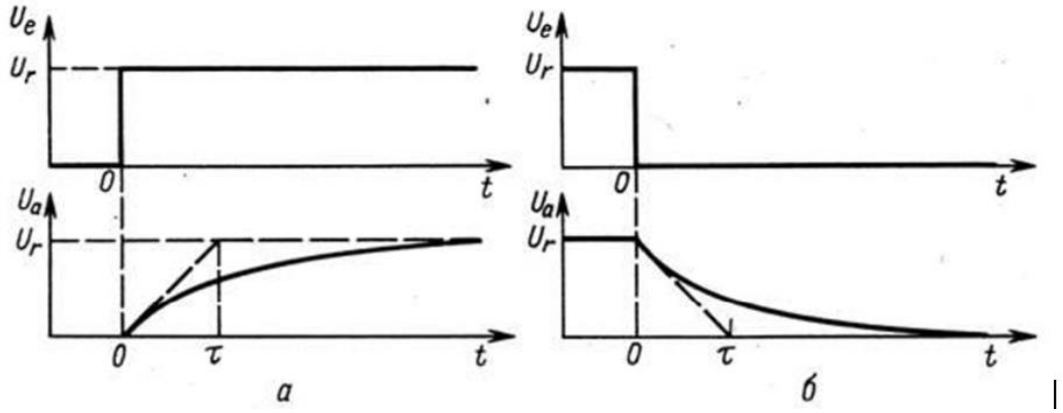


Рисунок 1.1.3 – Реакція ФНЧ на стрибок напруги

Щоб розрахувати вхідну напругу, використовується правило вузлів до ненавантаженого виходу. Тоді для схеми, що зображена на рис 1.1.1, розглянемо:

$$\frac{U_e - U_a}{R} - I_c = 0.$$

Враховуючи, що

$$I_c = C\frac{dU_a}{dt}$$

отримаємо диференціальне рівняння:

$$RC\dot{U}_a + U_a = U_e = \begin{cases} U_r \text{ при } t > 0 \\ 0 \text{ при } t > 0 \end{cases}$$

Диференціальне рівняння (1.5) має такий розв'язок:

– для рис. 1.1.3, а:

$$U_a(t) = U_r(1 - e^{-t/RC})$$

– для рис. 1.1.3, б:

$$U_a(t) = U_r e^{-t/RC}.$$

Відомо, що для встановлених значень Ua=Ur і Ua=0 криві будуть наближатися асимптотично. Тому в якості міри часу установки вихідної напруги прийнята постійна часу t. Вона показує інтервал часу, протягом якого процес досягає значення, що відрізняється від встановленого на 1/e (частину величини стрибка напруги на вході). З формул 1.1.6, 1.1.7 видно, що постійна часу дорівнює:

$$\tau = RC$$

Іншим параметром, що характеризує ФНЧ, є тривалість інтервалу фронту імпульсу. Цей параметр показує час, на протязі якого вихідна напруга зростає від 10 до 90 % кінцевого значення, якщо на вхід подати імпульс напруги прямокутної форми. Враховуючи властивості експоненціальної функції, з формул 1.6, 1.7 отримаємо:

$$t_a = t_{90\%} - t_{10\%} = \tau(\ln 0{,}9 - \ln 0{,}1) = \tau \ln 9 \approx 2{,}2\tau. \qquad (1.1.9)$$

При $f_g = 1/(2\pi\tau)$:

$$t_a \approx \frac{1}{3f_g}. \tag{1.1.10}$$

Це співвідношення з високим ступенем точності дійсне для ФНЧ. При послідовному з'єднанні декількох ФНЧ, що забезпечують різні тривалості фронту вихідного імпульсу t_{ai}, результуюча тривалість фронту імпульсу дорівнює:

$$t_a \approx \sqrt{\sum_i t_{ai}^2}. \tag{1.1.11}$$

Частота зрізу наближено визначається так:

$$f_g \approx \left(\sum_i f_{gi}^{-2}\right)^{-1/2}. \tag{1.1.13}$$

Фільтр верхніх частот

Фільтр верхніх частот (ФВЧ) – це схема, яка передає без змін сигнали високих частот, а на низьких частотах забезпечує затухання сигналів і випередження їх за фазою відносно вхідних сигналів. Схема простого RCфільтра верхніх частот наведена на рис. 1.1.4.

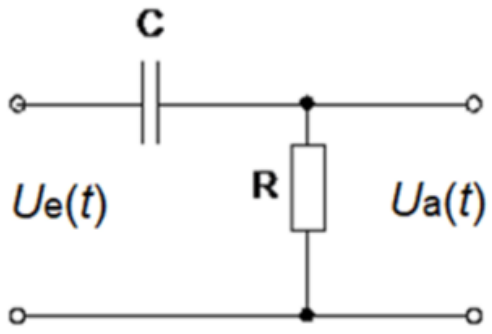


Рисунок 1.1.4 – Простий RC-фільтр верхніх частот

Амплітудно-частотні і фазочастотні характеристики отримаємо з формули для відношення напруг:

$$\underline{A}(j\omega)=\frac{\underline{U_a}}{\underline{U_e}}=\frac{R}{R+1/(j\omega C)}=\frac{1}{1+1/(j\omega RC)}. \qquad (1.1.14)$$

З виразу (1.14) визначається:

$$|\underline{A}|=\frac{1}{\sqrt{1+1/(\omega^2R^2C^2)}};\qquad \varphi=-arctg(1/(\omega RC)) \qquad (1.1.15)$$

Вираз для частоти зрізу співпадає з відповідним виразом для ФНЧ

$$f_g=\frac{1}{2\pi RC}. \qquad (1.1.16)$$

Фазовий зсув на цій частоті складає +45°. Як і для ФНЧ, найпростіше скласти АЧХ в подвійному логарифмічному масштабі за допомогою асимптот:

1) $|\underline{A}|=1\mathrel{\widehat{=}}0\partial Б$, на високих частотах f >>fg;

2) На низьких частотах f <<fg, відповідно до формули (1.1.15), $|A|\approx\omega RC$, тобто коефіцієнт підсилення пропорційний частоті. Нахил асимптоти дорівнює +20 дБ на декаду або +6дБ на октаву.

3) При f = fg, як і для ФНЧ, $|\underline{A}|=\frac{1}{\sqrt{2}\mathrel{\widehat{=}}}-3\partial Б,$

В результаті розрахунку реакції на імпульс напруги, постійна часу, як і для ФНЧ, дорівнює τ=RC.

При послідовному з’єднанні ФВЧ результуюча частота зрізу

$$f_g\approx\sqrt{\sum_i f_{gi}^2}. \qquad (1.1.17)$$

Контрольні питання.

1. Що таке «шум» у контексті даних або сигналу?
2. Назвіть два основні типи шуму, які можуть впливати на дані.
3. Який простий статистичний фільтр використовується для згладжування часового ряду?
4. Яка основна мета фільтрації даних?
5. Навіщо потрібно фільтрувати дані перед їх аналізом?

Лекція 4: Кластеризація та розпізнавання образів

- **4.1. Завдання кластеризації в автоматизації: виявлення аномалій, класифікація станів обладнання.**

Кластеризація, або кластерний аналіз — це статистична процедура, задача якої полягає в розбитті вибірки об'єктів на підмножини, що не перетинаються і називаються кластерами. Кожен кластер має складається зі схожих об'єктів, а об'єкти різних кластерів мають істотно відрізнятися один від одного. Задача кластеризації відноситься до статистичної обробки, а також до широкого класу задач навчання без вчителя. Ще її можна описати через задачу класифікації.

Задача кластеризації — це по факту задача класифікації, бо в обох випадках ми ділимо об'єкти на основі їх подібності між собою, але у випадку кластеризації приналежність навчальних об'єктів будь-яким класам не задається. Така задача — загальна, тому для її розв'язання використовуються різні підходи. Алгоритми побудови кластерів можуть дуже відрізнятися у підходах до того, що відносити в один кластер і як їх взагалі ефективніше шукати. Кластери можна утворювати ґрунтуючись на відстані між ними, на щільності ділянок у просторі даних, інтервалах або на конкретних статистичних розподілах. Це все залежать від конкретного набору даних та мети використання результатів. Кластерний аналіз не є автоматизованим, це скоріше ітераційний процес, тому що часто

доводиться змінювати метод опрацювання даних та параметри моделі, поки не буде отримано з результат з заданими властивостями.

Розв'язок неоднозначний, і на це є кілька причин. По-перше, не існує найкращого критерію якості кластеризації. Відомий цілий ряд досить ефективних критеріїв, а також ряд алгоритмів, які не мають чітко вираженого критерію, але все одно здійснюють досить якісну кластеризацію по побудові. Всі вони можуть давати різні результати. По-друге, число кластерів, як правило, не відомо заздалегідь і встановлюється відповідно до деякого суб'єктивного критерія. По-третє, результат кластеризації істотно залежить від метрики ρ, вибір якої, як правило, також суб'єктивний і визначається спеціалістом.

Щодо формальної постановки задачі:

Маємо вибірку $X\ell = \{x1, \ldots, x\ell\} \subset X$ і функцію відстані між об'єктами $\rho(x, x')$. Треба розбити вибірку на підмножини, які не будуть перетинатися і щоб кожен кластер складався з об'єктів, близьких по метриці ρ, а об'єкти різних кластерів істотно отличались. При цьому кожному об'єкту $xi \in X$ приписується мітка (номер) кластера yi.

Алгоритм кластеризації — це функція $a: X \to Y$, що будь-якому об'єкту $x \in X$ ставить у відповідність мітку кластера $y \in Y$. Множина міток Y в деяких випадках відома заздалегідь, однак

частіше завдання полягає в тому, щоб визначити оптимальне число кластерів з точки зору того чи іншого критерію якості кластеризації.

Задача групування набору об'єктів полягає в тому, що об'єкти в одному кластері більш схожі один на одного, ніж об'єкти в інших кластерах. Подібність — це буквально кількість, яка собою відображає міцність взаємозв'язку між двома об'єктами. Кластеризація використовується в основному для видобутка даних, а також в інших областях, таких як машинне навчання, розпізнавання образів, аналіз зображень, пошук інформації, біоінформатика, стиснення даних і комп'ютерна графіка.

Існує два типа кластеризації: жорстка та м'яка. У жорсткій кластеризації кожен об'єкт даних або повністю належить кластеру чи взагалі не належить. В м'якій кластеризації точка чи об'єкт даних може з певною ймовірністю належати більш ніж одному кластеру.

Типи вхідних даних та проблема міри відстані:

● Кожен об'єкт описується набором своїх характеристик, які називаються ознаками. Ознаки можуть бути числовими або нечисловими.

● Матриця відстаней між об'єктами. Кожен об'єкт описується відстанями до всіх інших об'єктів навчальної вибірки.

Дані, що використовуються в кластерному аналізі, можуть мати інтервальний, порядковий або категоріальний тип. Однак наявність суміші різних типів змінної зробить аналіз більш складним. Це

пояснюється тим, що в кластерному аналізі необхідно мати певний спосіб вимірювання відстані між спостереженнями, тобто тип використовуваної міри залежить від типу даних.

Для вимірювання відстані для бінарних та категоріальних даних було запропоновано ряд різних варіантів. Наприклад, для інтервальних даних найчастіше використовуваною мірою відстані є евклідова відстань.

Кластеризація — це досить суб'єктивна задача, для розв'язання якої може існувати більше одного правильного алгоритму. Кожен алгоритм слідує своєму набору правил для визначення «подібності» між об'єктами даних. Найбільш відповідний алгоритм кластеризації для конкретної проблеми часто потрібно вибирати експериментально, якщо немає математичної причини віддати перевагу одному алгоритму кластеризації над іншим. Алгоритм може добре працювати на певному наборі даних, але не працювати для іншого.

Алгоритми кластеризації можна класифікувати на основі кластерної моделі, яка ґрунтується на тому, як саме вони формують кластери або групи. Наприклад, такі як:

а) кластеризація на основі зв'язку: основна ідея кластеризації полягає в тому, що точки даних, які знаходяться ближче в просторі даних, є більш спорідненими (подібними). Кластери формуються шляхом з'єднання точок даних відповідно до їх відстані. На різних

відстанях будуть формуватися різні кластери, які можуть бути представлені за допомогою дендрограми (це також називають «ієрархічною кластеризацією»). Ці методи створюють саме ієрархію, а не унікальне розбиття набору даних, з якої користувачі все ще повинні обрати відповідні кластери, тобто рівень, на якому вони хочуть кластерувати. Вони також не дуже стійкі до викидів, які можуть з'являтися у вигляді додаткових кластерів або навіть призводити до злиття інших кластерів.

б) кластеризація на основі центроїда: у цьому типі кластери представлені центральним вектором або центроїдом. Цей центроїд не обов'язково має бути в наборі даних. Це ітеративні алгоритми кластеризації, в яких поняття подібності виводиться з того, наскільки близька точка даних до центроїда кластера. k-середніх, наприклад, є кластеризацією на основі центроїда.

Методи кластеризації

Методи, які ми розглянемо в рамках цієї роботи, є одними з найбільш популярних методів кластеризації — метод k-середніх та ієрархічний метод.

4.2 Метод k-середніх

Метод k-середніх є широко використовуваним методом кластеризації, який прагне мінімізувати середню квадратичну відстань між точками в одному кластері. Незважаючи на те, що він не гарантує абсолютної точності, його простота і швидкість це

компенсують. Доповнюючи метод k-середніх простою рандомізованою технікою висіву, отримуємо алгоритм, який є O(*log k*)-конкурентним з оптимальною кластеризацією. Експерименти показали, що таке доповнення досить сильно збільшує швидкість і точність k-середніх.

Перевагою методу k-середніх є те, що він більш зручний для кластеризації великої кількості спостережень, ніж, наприклад, метод ієрархічного кластерного аналізу, де дендограми швидко стають перевантаженими і мають тенденцію втрачати наочність.

Із недоліків методу можна відзначити те, що він чутливий до викидів, які можуть викривити середнє значення, та той факт, що кількість кластерів (що і являє тут собою k) має бути визначена спеціалістом заздалегідь.

У загальному сценарії кластеризація використовується у випадку, коли треба дізнатися більше про представлений набір даних. Таким чином є сенс запустити процес кластеризації не один раз, дослідити цікаві кластери і записати отримані дослідження. Кластеризація — це скоріше інструмент, який допомагає краще дослідити набір даних, але який не завжди повинен використовуватися як автоматичний метод класифікації даних. Вони бувають досить ненадійні, і одна кластеризація сама по собі не зможе дати всю інформацію, яку можна отримати з набору

даних.

Основна ідея методу k-середніх полягає у визначенні кластерів таким чином, щоб мінімальна сумарна варіація між кластерами (тобто загальна варіація в межах кластера) була мінімізована. Існує декілька алгоритмів для цього методу, однак стандартний алгоритм визначає загальну варіацію в межах кластера як суму квадратичних евклідових відстаней між елементами і відповідним центроїдом:

$$W(C_k) = \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

Де xi — точка даних, що належить до кластера C k , а μk — середнє значення точок, що присвоєні кластеру C k .

Кожне спостереження xi присвоюється заданому кластеру таким чином, щоб сума квадратів відстані спостереження до їх призначених кластерних центрів μk мінімізувалася.

Алгоритм виглядає так:

1. Визначити k — кількість кластерів, які будуть створені.

2. Вибрати випадкові k об'єктів з набору даних у ролі початкових центрів кластерів.

3. Призначити кожне спостереження найближчому центроїду на основі евклідової відстані між об'єктом і центроїдом.

4. Для кожного з k кластерів перерахувати центроїд кластера шляхом обчислення нового середнього значення усіх

точок даних у кластері.

5. Ітеративно мінімізувати загальну суму в межах площі. Повторити кроки 3 і 4, поки центроїди не зміняться або не буде досягнуто максимальної кількості ітерацій (R за замовчуванням використовує 10 як максимальну кількість ітерацій).

Загальна сума, або загальна варіація в межах кластера визначається таким чином:

$$\sum_{k=1}^{k} W(C_k) = \sum_{k=1}^{k} \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

4.3 Ієрархічна кластеризація

Ієрархічні алгоритми кластеризації, або алгоритми таксономії, будують не одне розбиття вибірки на непересічні класи, а систему вкладених розбиттів. Результат таксономії зазвичай представляється у вигляді таксономічного дерева — дендрограми. Класичним прикладом такого дерева є ієрархічна класифікація тварин і рослин.

Дендограми дозволяє уявити кластерну структуру у вигляді плаского графіка незалежно від того, яка розмірність початкового простору. Існують і інші способи візуалізації багатовимірних даних, такі як багатовимірне шкалювання або карти Кохонена, але вони привносять в картину штучні спотворення, вплив яких досить важко оцінити.

Є два типи методів:

1. Агломератні методи: нові кластери утворюються шляхом об'єднання дрібніших кластерів, і таким чином дерево створюється від листя до стовбура.

2. Дивізійні методи: нові кластери створюються шляхом ділення більших кластерів на більш дрібні, і таким чином дерево створюється від стовбура до листя.

Подібність кластерів часто розраховується через «неподібність», наприклад, евклідова відстань між двома кластерами. Отже, чим більше відстань між двома кластерами, тим краще.

Ключовою операцією в ієрархічній агломераційній кластеризації є неодноразове об'єднання двох найближчих кластерів у один кластер, але дуже важливо спочатку відповісти на три питання: як ви представити кластер з більш ніж однією точкою, як визначити «близькість» кластерів та коли перестати поєднувати кластери.

Злиття кластерів припиняється в залежності від доступної інформації про дані, які ми маємо. Якщо групувати футболістів на полі на основі їхніх позицій на полі, яке представлятиме їх координати для розрахунку відстані між гравцями, очевидно, що треба зупинитися на лише двох кластерах, оскільки можуть бути тільки дві команди, які грають у футбольний матч.

Алгоритм методу виглядає таким чином:

1. Обчислити матрицю близькості, що містить відстань між кожною парою шаблонів. Розглядати кожен зразок як окремий кластер.

2. Знайти найбільш схожу пару кластерів за допомогою матриці близькості. Об'єднати ці два кластера в один більший кластер. Оновити матрицю близькості, щоб відобразити цю операцію злиття.

3. Якщо всі шаблони знаходяться в одному кластері, зупинитися. В іншому випадку перейти до кроку 2.

Ієрархічний алгоритм дає дендограму, що представляє собою складене групування шаблонів і рівні схожості, на яких змінюються самі групування.

Більшість ієрархічних алгоритмів кластеризації є варіантами однозв'язного і повнозв'язного алгоритму, а також алгоритму мінімальної дисперсії. Найбільш популярними з них є однозв'язний та повнозв'язний алгоритми. Вони відрізняються тим, як вони характеризують подібність між парою кластерів.

У методі з однозв'язним алгоритмом відстань між двома кластерами — це мінімум відстаней між усіма парами шаблонів, взятих з двох кластерів (один з першого кластера, другий — з другого).

У повнозв'язному алгоритмі відстань між двома кластерами — це максимум усіх попарних відстаней між шаблонами в двох

кластерах. В обох випадках два кластери об'єднуються для формування більшого кластера на основі мінімальних критеріїв відстані. Повнозв'язний алгоритм створює щільні та компактні кластери, а однозв'язний алгоритм, навпаки, страждає від ланцюгового. Він має тенденцію виробляти кластери, які є занадто громіздкими або подовженими.

Дослідження набору даних методами кластеризації

Спробуємо дослідити набір даних розглянутими вище методами кластеризації. Використаємо набір даних під назвою «Іриси Фішера», на прикладі якого Рональд Фішер продемонстрував роботу розробленого ним методу дискримінантного аналізу. Також називають ірисами Андерсона, так як дані були зібрані американським ботаніком Едгаром Андерсоном. Цей набір даних вважається класичним і часто використовується в літературі для ілюстрації роботи різних статистичних алгоритмів.

Для дослідження будемо використовувати середу розробки R. Поглянемо на дані:

```
library(datasets)
data(iris) str(iris)
summary(iris)
```

```
> str(iris)
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
```

Цей відомий набір даних про іриси дає вимірювання в сантиметрах по довжині та ширині чашолистка та довжині та ширині пелюстки відповідно для 50 квітів кожного з 3 видів ірису. Видами є *iris setosa*, *versicolor*та *virginica*.

Ще трохи поглянемо на набір даних. Подивимося, які наявні кореляції між змінними.

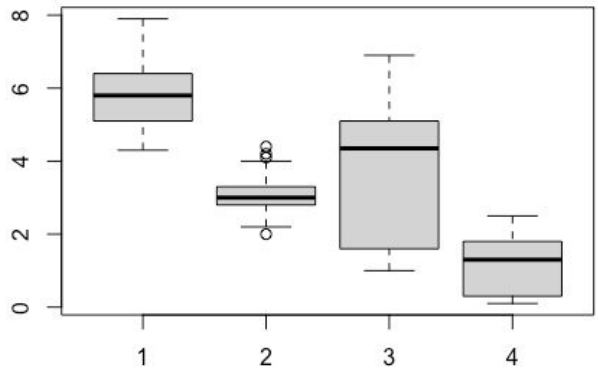
```
> cor(iris[1:4])
             Sepal.Length Sepal.Width Petal.Length Petal.Width
Sepal.Length    1.0000000  -0.1175698    0.8717538   0.8179411
Sepal.Width    -0.1175698   1.0000000   -0.4284401  -0.3661259
Petal.Length    0.8717538  -0.4284401    1.0000000   0.9628654
Petal.Width     0.8179411  -0.3661259    0.9628654   1.0000000
```

алекці

Бачимо, що кореляції присутні і досить значні. Наприклад, 87% кореляції між довжинами пелюсток і чашолистків та 96% між шириною та довжиною пелюстків.

Також подивимося на наявність очевидних викидів в даних. Бачимо, що нічого критичного, особливих викидів немає, лише трохи в змінній довжині чашолистка.

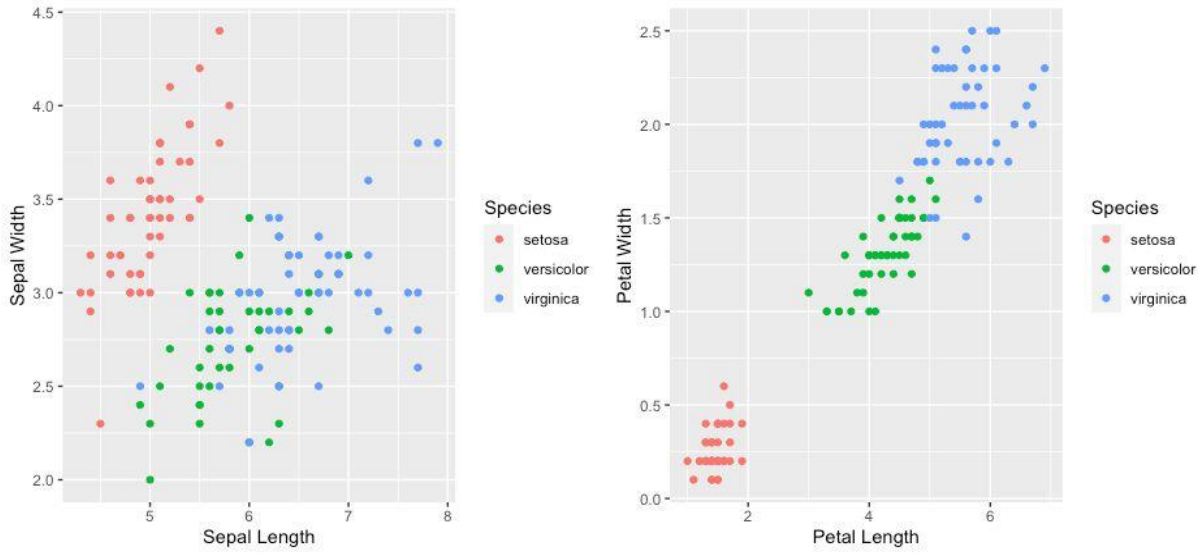
```
boxplot(iris$Sepal.Length, iris$Sepal.Width,
iris$Petal.Length, iris$Petal.Width)
```



Графіки довжини і ширини пелюсток та чашолистків:

```
g1 <- ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width, color =
Species)) +
        geom_point() +
        labs(x = "Sepal Length", y = "Sepal Width")
g2 <- ggplot(iris, aes(x = Petal.Length, y = Petal.Width, color =
Species)) +
        geom_point() +
        labs(x = "Petal Length", y = "Petal Width")
```

grid.arrange(g1, g2, nrow = 1)



Перейдемо до методу k-середніх.

```
set.seed(99)
irisCl <- kmeans(iris[,1:4], center = 3, nstart = 20)
irisCl
clusplot(iris, irisCl$cluster, color=T, labels=0, lines=0)
```

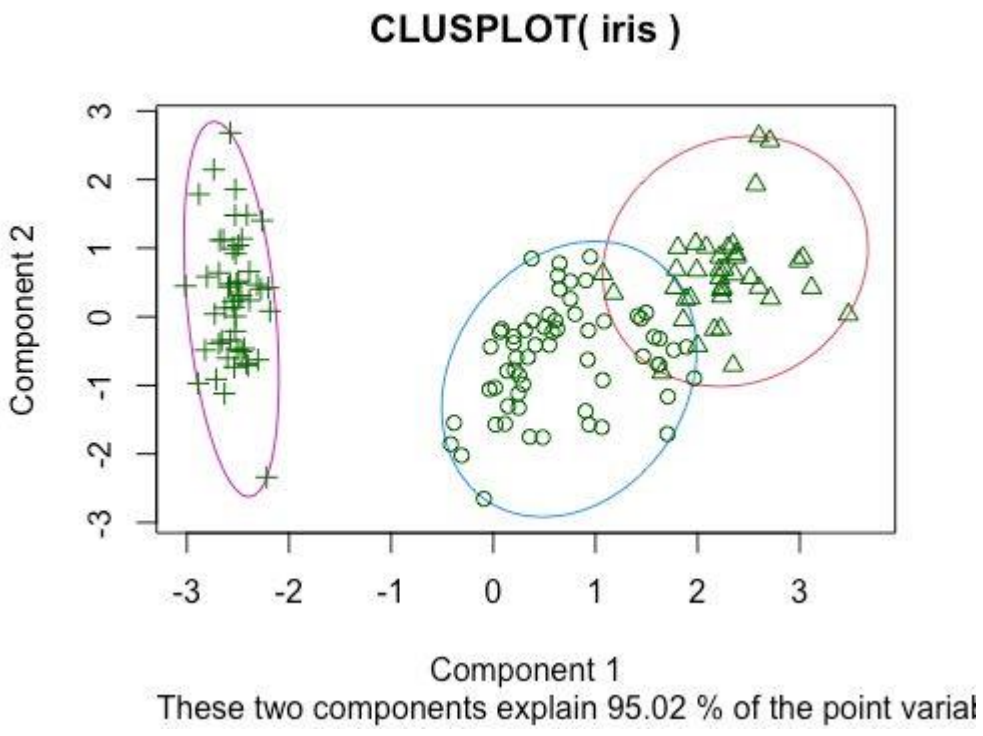
```
> irisCl
K-means clustering with 3 clusters of sizes 62, 38, 50

Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1     5.901613    2.748387     4.393548    1.433871
2     6.850000    3.073684     5.742105    2.071053
3     5.006000    3.428000     1.462000    0.246000

Clustering vector:
  [1] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
 [45] 3 3 3 3 3 3 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1
 [89] 1 1 1 1 1 1 1 1 1 1 1 2 1 2 2 2 2 1 2 2 2 2 2 2 1 1 2 2 2 2 1 2 1 2 1 2 2 1 1 2 2 2 2
[133] 2 1 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 1

Within cluster sum of squares by cluster:
[1] 39.82097 23.87947 15.15100
 (between_SS / total_SS =  88.4 %)
```

Кластерів, ми знаємо, три, тому *center = 3*, маємо три кластери з такою кількістю об'єктів: 62, 38, 50. Графік:



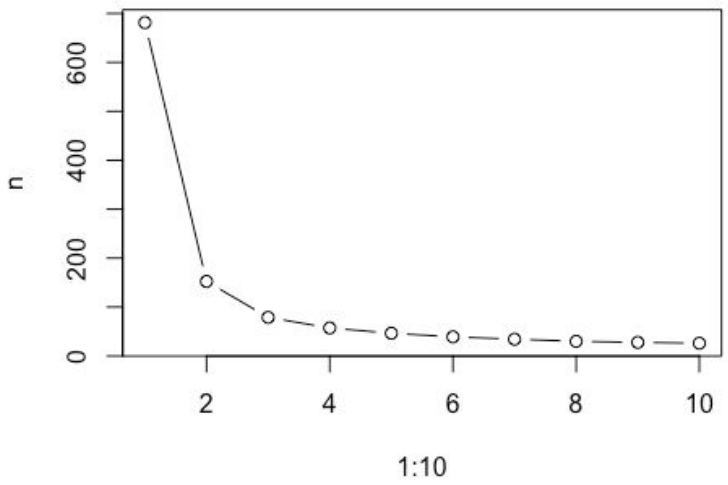
Видно, що кластеризація дозволила пояснити 95.02% даних, *setosa* повністю чітко відображена, а між *virginica* та *versicolor* є

деякий шум. Власне, можемо побачити таблицю з правильністю:

	setosa	versicolor	virginica
1	0	48	14
2	0	2	36
3	50	0	0

Ще розглянемо варіант, коли нам невідома кількість кластерів і нам треба самим оцінити оптимальну кількість, що ми зробимо «ліктьовим методом».

```
n <- vector(mode = "character", length = 10)
for (i in 1:10){
irisCluster <- kmeans(iris[,1:4], center = i, nstart = 20)
n[i] <- irisCluster$tot.withinss}
plot(1:10, n, type = "b", pch = 1)
```



Три кластери і є оптимальною кількістю.

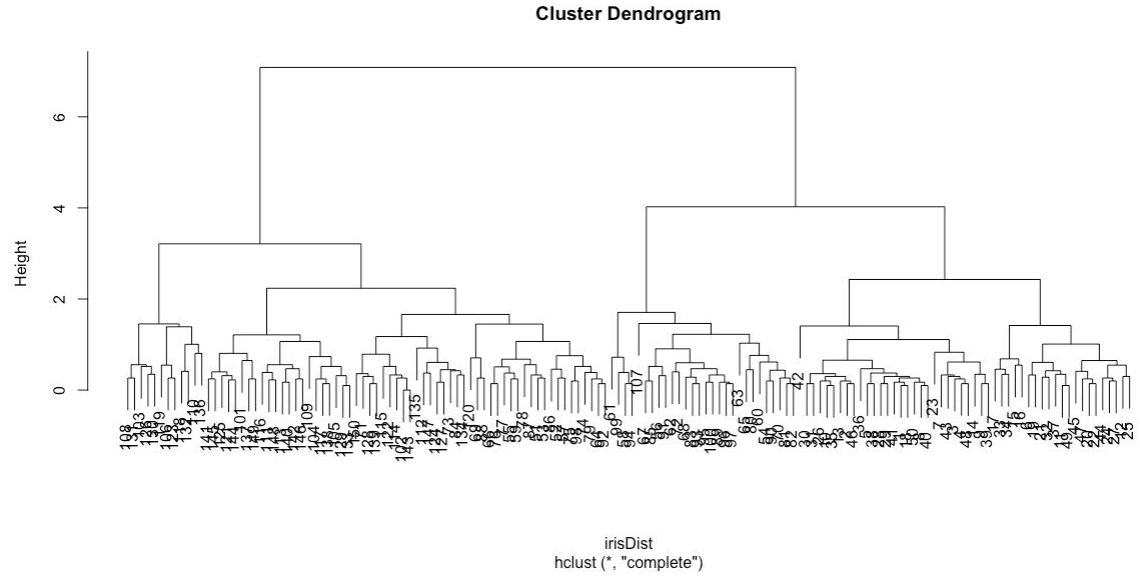
Щодо ієрархічного методу:

```
set.seed(100)
species <- as.list(iris$Species)
species <- unlist(species)
irisDist <- dist(iris[1:4], method = "euclidean") model
<- hclust(irisDist, method = "complete") modelcut <-
cutree(model, 3)
```

```
table(species, modelcut)
```

```
> table(species, modelcut)
            modelcut
species       1  2  3
  setosa     50  0  0
  versicolor  0 23 27
  virginica   0 49  1
```

Дендрограма виглядає таким чином:



Оцінка результатів і висновки

У цій роботі ми розглянули такі методи кластеризації, як метод k-середніх та ієрархічний метод. Метод k-середніх дуже широко використовується, він прагне мінімізувати середню квадратичну відстань між точками в одному кластері, а ієрархічний метод будує систему вкладених розбиттів. В нашому випадку ми використовували повнозв'язний алгоритм, де відстань між двома кластерами — це
максимум усіх попарних відстаней між шаблонами в двох кластерах.

Ми розглянули ці методі на прикладі класичного набору даних — ірисах Фішера. У випадку методу k-середніх ми обрали ліктьовим методом оптимальну кількість кластерів (три), а потім застосували безпосередньо кластеризацію і отримали точність

88,4%, а розбиття на три кластери дозволило пояснити 95% варіації.

При ієрархічному методі, як і при методі k-середніх, вид *setosa* був чітко визначений на відміну від інших двох через те, що вони в деяких місцях досить схожі, але незважаючи на це, ми побачили, що точність була досить висока, хоч найменш точно було розпізнано вид *versicolor*.

- **4.4. Застосування в технічному зорі: сегментація зображень, виявлення об'єктів.**

Вибрати метод кластеризації і на його основі побудувати і реалізувати алгоритм сегментації досить просто. Велика частина літератури з сегментації зображень - це роботи по кластеризації (хоча про це не завжди говориться прямо).

У технічному зорі зображення представлене як набір пікселів. Кластеризація дозволяє групувати ці пікселі на основі їхніх характеристик (кольору, яскравості, текстури, позиції), щоб виявити осмислені області без необхідності попередніх міток для навчання.

Сегментація зображень (Image Segmentation)

Як використовується кластеризація?

Алгоритм розглядає кожен піксель як точку даних у багатовимірному просторі. Вимірності цього простору можуть включати:

- **Кольорові канали:** (R, G, B) або (L, a, b).
- **Просторові координати:** (x, y) положення пікселя.
- **Текстура:** характеристики, отримані за допомогою фільтрів.

Популярні алгоритми та підходи:

а) K-Means для сегментації за кольором:

- **Процес:**

1. Усі пікселі зображення представляються у вигляді точок у, наприклад, 5-вимірному просторі (R, G, B, x, y).
2. Запускається алгоритм K-Means, який групує пікселі в K кластерів.
3. Кожному пікселю присвоюється колір центроїда його кластера.

- **Результат:** Зображення, розбите на K кольорових областей.
- **Недолік:** Може не враховувати просторову зв'язність, що призводить до "плямистості".

б) Mean Shift сегментація:

- **Принцип:** Цей алгоритм не потребує завдання кількості кластерів. Він знаходить "зони притягання" в просторі даних, які природним чином відповідають сегментам на зображенні.
- **Перевага:** Краще працює з кластерами довільної форми та автоматично визначає їхню кількість.
- **Результат:** Часто більш "природно" виглядаючі сегменти порівняно з K-Means.

Приклад застосування:

- Виділення неба, лісу, води на супутникових знімках.
- Медична візуалізація: виділення пухлин або органів на МРТ/КТ знімках.

Виявлення об'єктів (Object Detection)

Роль кластеризації у виявленні об'єктів:

Безпосередньо для фінального виявлення сьогодні використовують складніші моделі (на основі глибокого навчання), однак кластеризація грає вирішальну роль у двох ключових етапах:

а) Генерація регіон-пропозицій (Region Proposal):

- **Завдання:** Швидко знайти області на зображенні, де з високою ймовірністю може знаходитися об'єкт, щоб не перебирати всі можливі вікна.
- **Як це працює?** Алгоритми на кшталт **Selective Search** використовують кластеризацію подібних за кольором і текстурою сусідніх пікселів, щоб сформувати ієрархію регіонів різного розміру. Ці регіони і є пропозиціями, які потім подаються на вхід класифікатору.

б) Кластеризація в якості ансамблю (для підвищення якості):

- **Завдання:** Сучасні детектори (наприклад, сімейство YOLO) часто прогнозують дуже багато bounding box'ів навколо одного об'єкта.
- **Як це працює?** Алгоритм кластеризації, такий як **DBSCAN** або простий агломеративний, використовується для групування цих перекриваючих прямокутників, які вказують на один і той же об'єкт. Після цього з кожного кластера прямокутників вибирається найкращий (найбільш "впевнений") або вони об'єднуються в один.

в) Кластеризація анкор-боксів (Anchor Boxes):

- **Завдання:** У таких архітектурах, як YOLO чи SSD, необхідно заздалегідь визначити набір "типових" розмірів та форм об'єктів (anchor boxes) для навчання.
- **Як це працює?** Алгоритм **K-Means** застосовується до розмірів (width, height) об'єктів у тренувальному наборі даних. Він знаходить K найпоширеніших розмірів, які потім використовуються як анкори, що значно покращує ефективність детектора.

Контрольні запитання.

1. Що таке основна мета кластеризації даних?
2. Яка ключова відмінність між кластеризацією та класифікацією?
3. Назвіть два популярні алгоритми кластеризації.
4. Який з алгоритмів — K-means чи DBSCAN — не вимагає від нас заздалегідь вказувати кількість кластерів?
5. Як можна використати кластеризацію для виявлення аномалій (викидів) у даних?

Лекція 5: Системи зберігання інформації. Бази даних

5.1. Роль баз даних (БД) в інтегрованих системах.

Від початку розвитку обчислювальної техніки утворилися два основних напрямки її використання. Перший передбачав застосування обчислювальної техніки для виконання чисельних розрахунків, які складно або й взагалі неможливо здійснити вручну. Становлення цього напрямку сприяло інтенсифікації методів числових розв'язків складних математичних задач та розвитку класу мов програмування, орієнтованих на зручний запис алгоритмів. Другий напрямок окреслював шляхи використання засобів обчислювальної техніки для автоматичної обробки даних. Саме цей напрямок і є предметом вивчення дисципліни "Бази даних та інформаційні системи". Бази даних, системи управління базами даних, системи обробки даних У процесі свого розвитку людство століттями накопичувало знання. І лише у другій половині XX та на початку XXI століття надзвичайної актуальності набуло питання сортування та обробки всієї наявної інформації. Зберігання та обробка інформації – найважливіші функції комп'ютера. Саме в процесі розв'язування конкретних прикладних задач відбувається обробка потрібних даних за заданим алгоритмом. Дані можуть бути найрізноманітнішими: числа, прізвища, адреси, назви тощо. Точкою відліку нової інформаційної епохи стала саме поява перших баз

даних (у подальшому – БД). У найзагальнішому випадку БД – це файл спеціального формату, що містить певним чином структуровану інформацію.

БД – великий упорядкований комплекс інформації, який зберігається на комп'ютерних носіях зазвичай разом із програмою, що дозволяє здійснювати швидкий пошук даних, їх оновлення та друк.

Без власної БД тепер не обходиться жоден навчальний заклад, державна установа, приватна фірма чи корпорація.

Сучасні технології БД є результатом розвитку протягом кількох останніх десятиліть способів обробки даних та керування наявною інформацією. Обробка даних розвивалася від примітивних методів 50-х років до складних інтегрованих систем сьогодення. Перші системи обробки даних виконували лише канцелярську роботу, скорочуючи тим самим паперовий обіг. Новіші системи накопичували інформацію та здійснювали управління нею. Сьогодні для роботи з файлами баз даних використовують спеціальне програмне забезпечення – **системи управління базами даних (СУБД)**. База даних орієнтована на інтегровані запити, а не на одну програму, і використовується для інформаційних потреб багатьох користувачів. В зв'язку з цим бази даних дозволяють в значній мірі скоротити надлишковість інформації. Перехід від структури БД до

потрібної структури в програмі користувача відбувається автоматично за допомогою СУБД.

Засобами СУБД здійснюється наповнення бази даних, пошук необхідної інформації, отримання звітів, створення нових або видалення існуючих БД, експортування та імпортування даних тощо. Фактично під базою даних частіше розуміють не саму базу даних як сховище інформації, а й супровідне програмне забезпечення, тобто СУБД+БД. Перші комп'ютери використовувалися в основному в комерційних структурах для ведення бухгалтерського обліку, створення відомостей зарплатні тощо. Витрати ручної праці для ведення подібних відомостей або виписуванню рахунків були настільки значними, що автоматична система, яка могла б виконувати ці функції, швидко окупалася. Оскільки така система виконувала звичайні функції роботи з документами (які досі виконувалися вручну), використовуючи для цього дані комерційної структури, вона отримали назву системи обробки даних. Комп'ютерні файли виконували роль звичайних па8 пок для паперів і містили певну інформацію. **Система обробки даних – автоматична система для роботи з даними установи.**

СУБД – це складна програмна система накопичення та з наступним маніпулюванням даними, що представляють інтерес для користувача. Кожній прикладній програмі СУБД надає інтерфейс з базою даних та має засоби безпосереднього доступу до неї. Таким

чином, СУБД відіграє центральну роль в функціонуванні автоматизованого банку даних.

Архітектурно СУБД складається з двох великих компонент (рис.1.1). За допомогою мови опису даних (МОД) створюються описи елементів, груп та записів даних, а також взаємозв'язки між ними, які, як правило, задаються у вигляді таблиць. В залежності від конкретної реалізації СУБД мову опису даних підрозділяють на мову опису схеми бази даних (МОС) та мову опису підсхем бази даних (МОП). Слід особливо зазначити, що МОД дозволяє створити не саму базу даних, а лише її опис.

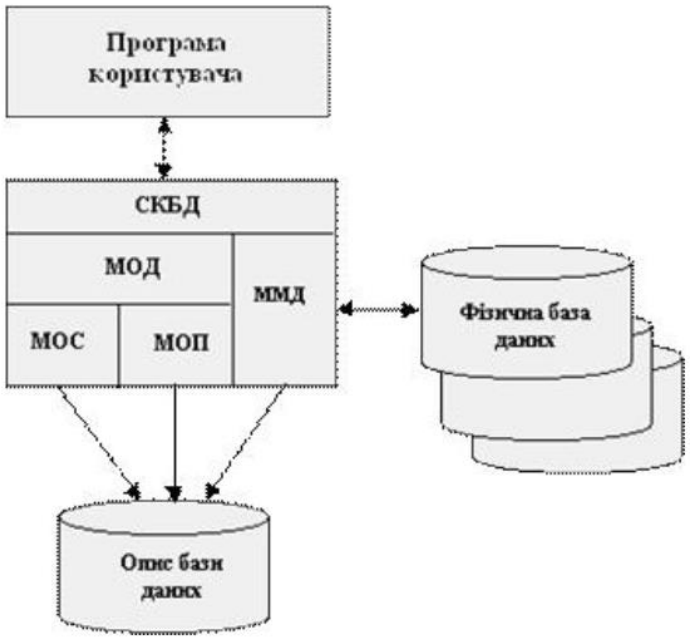


Рисунок 1.1 – Архітектура СУБД

Для виконання операцій з базою даних в прикладних програмах використовується мова маніпулювання даними (ММД). Фактична

структура фізичного зберігання даних відома тільки СУБД.

З метою забезпечення зв'язків між програмами користувачів і СУБД (що особливо важливо при мультипрограмному режимі роботи операційної системи) в СУБД виділяють особливу складову – резидентний модуль системи керування базами даних. Цей модуль значно менший від всієї СУБД, тому на час функціонування автоматизованого банку інформації він може постійно знаходитись в основній пам'яті ЕОМ та забезпечувати взаємодію всіх складових СУБД і програм, які до неї звертаються.

Приведена структура притаманна усім СУБД, котрі розрізняються обмеженнями та можливостями по виконанню відповідних функцій. Отже, процес порівняння і оцінки таких систем для одного конкретного застосування зводиться до співставлення можливостей наявних СУБД з вимогами користувачів.

До недавнього часу при організації обробки інформації на ЕОМ застосовувався підхід, при якому на основі інформації одного і того ж об'єкту управління (наприклад, матеріальних ресурсів) в залежності від її вигляду (нормативна. розцінкова тощо) і ступеню постійності формувались масиви лінійної структури двох типів: умовно-постійні (з інформацією, яка використовувалась багато разів протягом довгого часу) і умовно-перемінні (з фактичною або поточною інформацією). Створення і багаторазове використання масивів з умовно-постійною інформацією має ті переваги, які

дозволяють значно спростити первинну документацію шляхом виведення з її складу ряд постійних реквізитів. знизити трудомісткість робіт на стадії заповнення первинних документів, підготовки і вроду фактичної або поточної інформації до ЕОМ. Недоліком таких масивів, які мають лінійну структуру, є то що інформація одного і того я об'єкту управління розосереджується поміж багатьох різних масивів (нормативних, планових та ін.), що неминуче веде до дублювання деяких реквізитів, ускладненню при спільній їх обробці тощо, а головне – не дає змоги реалізувати принцип незалежності від прикладних програм користувача. Лінійні масиви, сформовані традиційним способом, ефективні, як правило, а позиції одного застосування.

З розвитком інформаційного забезпечення систем автоматизованої обробки інформації, прагненням забезпечити виконання нових режимів обробки даних у реальному часі і з мультидоступом до схованих даних позначилась нова тенденція до складення інформаційного забезпечення розподілених баз даних. В умовах використання таких баз створюються комплексні масиви нелінійної структури, які мають усі дані про ту чи іншу предметну область або про керований об'єкт як постійного, так і перемінного характеру.

5.2. Моделі даних: ієрархічна, мережна, реляційна.

Моде́ль да́них (англ. Data model) — абстрактне представлення реального світу, що відображає тільки ті об'єкти, що безпосередньо

стосуються програми. Це, як правило, визначає специфічну групу об'єктів, їх атрибутивне значення і відношення між ними. В випадку ГІС, використовується механізм представлення і організації просторової моделі даних, або растрової моделі даних. Вона не залежить від комп'ютерної системи і пов'язана тільки з структурою даних. Основою бази даних є модель даних — фіксована система понять і правил для представлення даних структури, стану і динаміки проблемної області в базі даних. У різний час послідовне застосування одержували ієрархічна, мережна і реляційна моделі даних. У наш час усе більшого поширення набуває об'єктно-орієнтований підхід до організації баз даних ГІС.

Ієрархічна модель даних. Ієрархічні бази даних можуть бути представлені як дерево, що складається з об'єктів різних рівнів. Верхній рівень займає один об'єкт, другий - об'єкти другого рівня і т. д. Між об'єктами існують зв'язки, кожен об'єкт може включати в себе декілька об'єктів нижчого рівня. Такі об'єкти перебувають у відношенні предка (об'єкт більш близький до кореня) до нащадка (об'єкт більш низького рівня), при цьому можлива ситуація, коли об'єкт-предок не має нащадків або має їх кілька, тоді як в об'єкта-нащадка обов'язково тільки один предок. Об'єкти, що мають загального предка, називаються близнюками. Наприклад, якщо ієрархічна база даних містила інформацію про покупців та їх

замовленнях, то буде існувати об'єкт "покупець" (батько) і об'єкт "замовлення" (дочірній). Об'єкт "покупець" матиме покажчики від кожного замовника до фізичного розташування замовлень покупця на об'єкт "замовлення".

У цій моделі запит, направлений вниз по ієрархії, простий (наприклад: які замовлення належать цьому покупцеві), а проте запит, направлений вгору по ієрархії, більш складний (наприклад, який покупець помістив це замовлення). Також, важко уявити не-ієрархічні дані при використанні цієї моделі.

Ієрархічної базою даних є файлова система, що складається з кореневої директорії, в якій є ієрархія піддиректорій і файлів. **Ієрархічна модель даних.** Перші системи управління базами даних використовували ієрархічну модель даних, і в часі їх поява передує появі мережевої моделі.

Структурна частина ієрархічної моделі.

Основними інформаційними одиницями в ієрархічній моделі даних є сегмент і поле. Поле даних визначається як найменша неподільна одиниця даних, доступна користувачеві. Для сегмента визначаються тип сегмента та примірник сегмента. Примірник сегмента утвориться з конкретних значень полів даних. Тип сегмента - це пойменована сукупність вхідних у нього типів полів даних. Як і мережева, ієрархічна модель даних базується на графових формі побудови даних, і на концептуальному рівні вона є

просто окремим випадком мережевої моделі даних. В ієрархічній моделі даних вершині графа відповідає тип сегмента або просто сегмент, а дуг - типи зв'язків предок - нащадок. В ієрархічних структурах сегмент - нащадок повинен мати в точності одного предка.

Ієрархічна модель являє собою зв'язний неорієнтований граф деревовидної структури, що об'єднує сегменти. Ієрархічна БД складається з упорядкованого набору дерев.

Перетворення концептуальної моделі в ієрархічну модель даних. Перетворення концептуальної моделі в ієрархічну структуру даних багато в чому схоже з перетворенням її в мережеву модель, але й має деякі відмінності у зв'язку з тим, що ієрархічна модель вимагає організації всіх даних у вигляді дерева. Перетворення зв'язку типу "один до багатьох" між предком і нащадком здійснюється практично автоматично в тому випадку, якщо нащадок має одного предка, і відбувається це таким чином. Кожен об'єкт з його атрибутами, що бере участь в такого зв'язку, стає логічним сегментом. Між двома логічними сегментами встановлюється зв'язок типу "один до багатьох". Сегмент з боку "багато" стає нащадком, а сегмент з боку "один" стає предком.

Ситуація значно ускладнюється, якщо нащадок в зв'язку має не одного, а двох і більше предків. Так як подібне положення є неможливим для ієрархічної моделі, то відображена структура

даних потребує перетвореннях, які зводяться до заміни одного дерева, наприклад, двома (якщо є два предка). В результаті такого перетворення в базі даних з'являється надмірність, так як єдино можливий вихід з цієї ситуації - дублювання даних.

Керуюча частина ієрархічної моделі. У рамках ієрархічної моделі виділяють мовні засоби опису даних (ЯОД) та засоби маніпулювання даними (ММД). Кожна фізична база описується набором операторів, що обумовлюють як її логічну структуру, так і структуру зберігання БД. При цьому спосіб доступу встановлює спосіб організації взаємозв'язку фізичних записів.

Визначено такі способи доступу:

ієрархічно послідовний;

ієрархічно індексного-послідовний;

ієрархічно прямій;

ієрархічно індексного-прямій;

індексний.

Крім завдання імені БД і способу доступу опису повинні містити визначення типів сегментів, що складають БД, відповідно до ієрархії, починаючи з кореневого сегмента. Кожна фізична БД містить тільки один кореневий сегмент, але в системі може бути декілька фізичних БД.

Серед операторів маніпулювання даними можна виділити

оператори пошуку даних, оператори пошуку даних з можливістю модифікації, оператори модифікації даних. Набір операцій маніпулювання даними в ієрархічній БД невеликий, але цілком достатній.

Приклади типових операторів пошуку даних знайти вказане дерево БД;

перейти від одного дерева до іншого;

знайти примірник сегмента, що задовольняє умовам пошуку;

перейти від одного сегмента до іншого всередині дерева;

перейти від одного сегмента до іншого в порядку обходу ієрархії.

Приклади типових операторів пошуку даних з можливістю модифікації:

знайти і утримати для подальшої модифікації єдиний екземпляр сегмента, що задовольняє умовам пошуку;

знайти і утримати для подальшої модифікації наступний примірник сегменту з тими ж умовами пошуку;

знайти і утримати для подальшої модифікації наступний примірник для того ж батька.

Приклади типових операторів модифікації ієрархічно організованих даних, які виконуються після виконання одного з операторів другої групи (пошуку даних з можливістю модифікації):

вставити новий екземпляр сегмента в зазначену позицію;

оновити поточний екземпляр сегмента;

видалити поточний екземпляр сегмента.

В ієрархічній моделі автоматично підтримується цілісність посилань між предками і нащадками. Основне правило: ніякої нащадок не може існувати без свого батька.

Відомі ієрархічні СУБД. Типовим представником (найбільш відомим і поширеним) є Information Management System (IMS) фірми IBM.

Перша версія з'явилася в 1968 р.

Time-Shared Date Management System (TDMS) компанії Development Corporation;

Mark IV Multi - Access Retrieval System компанії Control Data Corporation; System 2000 розробки SAS-Institute;

Сервери каталогів, такі, як LDAP і Active Directory (допускають чітке уявлення у вигляді дерева).

За принципом ієрархічної БД побудовані ієрархічні файлові системи та Реєстр Windows.

InterSystems Cach

Google App Engine DataStore API.

Мережна модель даних. Мережева модель даних - логічна модель даних, яка є розширенням ієрархічного підходу, строга математична теорія, що описує структурний аспект, аспект

цілісності й аспект обробки даних у мережевих базах даних.

Різниця між ієрархічної моделлю даних і мережевої полягає в тому, що в ієрархічних структурах запис-нащадок повинна мати в точності одного предка, а в мережевій структурі даних у нащадка може матися будь-яке число предків.

Мережева БД складається з набору примірників певного типу запису і набору примірників певного типу зв'язків між цими записами.

Тип зв'язку визначається для двох типів запису: предка і нащадка. Примірник типу зв'язку складається з одного примірника типу запису предка і впорядкованого набору примірників типу запису нащадка. Для даного типу зв'язку L з типом запису предка P і типом запису нащадка C повинні виконуватися наступні дві умови:

кожен екземпляр типу запису P є предком тільки в одному екземплярі типу зв'язку L;

кожен екземпляр типу запису C є нащадком не більше ніж в одному примірнику типу зв'язку L.

Аспект маніпуляції. Зразковий набір операцій маніпулювання даними:

знайти певний запис в наборі однотипних записів;

перейти від предка до першого нащадку за деякою зв'язку;

перейти до наступного нащадку у деякому зв'язку;

перейти від нащадка до предка по деякому зв'язку;

створити новий запис;

знищити запис;

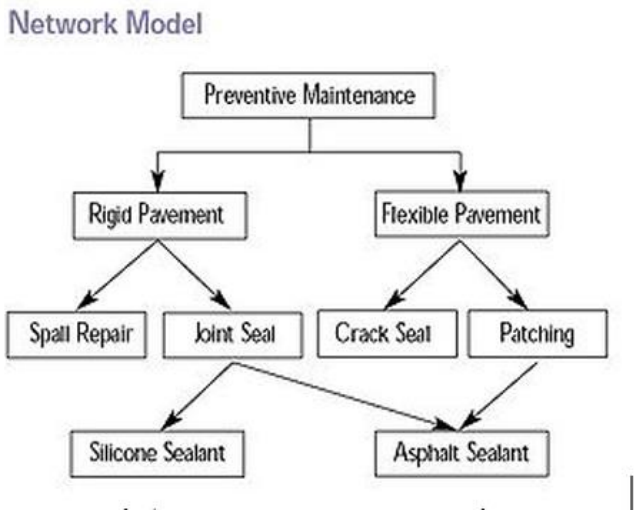
модифіковані запис;

включити у зв'язок;

виключити із зв'язку;

переставити в інший зв'язок і т. д.

Аспект цілісності. Є (необов'язкова) можливість вимагати для конкретного типу зв'язку відсутність нащадків, не беруть участь ні в одному примірнику цього типу зв'язку (як в ієрархічній моделі). Переваги. Перевагою мережевої моделі даних є можливість ефективної реалізації за показниками витрат пам'яті й оперативності. **Недоліки** Недоліком мережевої моделі даних є висока складність і жорсткість схеми БД, побудованої на її основі. Мережева СУБД, графічне представлення зв'язків.



До основних понять мережевої моделі бази даних належать: рівень, елемент (вузол), зв'язок.

Вузол - це сукупність атрибутів даних, що описують деякий об'єкт. На схемі ієрархічного дерева вузли представляються вершинами графа. У мережній структурі кожен елемент може бути пов'язаний з будь-яким іншим елементом.

Мережні бази даних подібні ієрархічним, за винятком того, що в них є покажчики в обох напрямках, які з'єднують споріднену інформацію. Незважаючи на те, що ця модель вирішує деякі проблеми, пов'язані з ієрархічною моделлю, виконання простих запитів залишається досить складним процесом.

Також, оскільки логіка процедури вибірки даних залежить від фізичної організації цих даних, то ця модель не є повністю

незалежною від програми. Іншими словами, якщо необхідно змінити структуру даних, то потрібно змінити і додаток.

Приклади мережевих СУБД: СООБЗ Cerebrum [1] ІСУБД CronosPRO [2] dbVista Cach GT.M.

Реляційна модель та її характеристики. Реляційна модель даних (РМД) - логічна модель даних, прикладна теорія побудови баз даних, яка є додатком до завдань обробки даних таких розділів математики як теорії множин і логіка першого порядку.

На реляційної моделі даних будуються реляційні бази даних.

Реляційна модель даних включає такі компоненти:

Структурний аспект (складова) - дані в базі даних є набором відносин.

Аспект (складова) цілісності - відносини (таблиці) відповідають певним умовам цілісності.

РМД підтримує декларативні обмеження цілісності рівня домену (типу даних), рівня відносини і рівня бази даних.

Аспект (складова) обробки (маніпулювання) - РМД підтримує оператори маніпулювання відносинами (реляційна алгебра, реляційне числення).

Крім того, до складу реляційної моделі даних включають теорію нормалізації.

Термін "реляційний" означає, що теорія заснована на математичному понятті ставлення (relation). Як неформального синоніма терміну "відношення" часто зустрічається слово таблиця. Необхідно пам'ятати, що "таблиця" є поняття нестроге і неформальне і часто означає не "ставлення" як абстрактне поняття, а візуальне уявлення відносини на папері або екрані. Некоректне і нестрогое використання терміну "таблиця" замість терміна "ставлення" нерідко призводить до нерозуміння. Найбільш часта помилка полягає в міркуваннях про те, що РМД має справу з "плоскими", або "двовимірними" таблицями, тоді як такими можуть бути тільки візуальні представлення таблиць. Відносини ж є абстракціями, і не можуть бути ні "плоскими", ні "неплоским".

Для кращого розуміння РМД слід відзначити три важливі обставини: модель є логічною, тобто відносини є логічними (абстрактними), а не фізичними (збереженими) структурами; для реляційних баз даних вірний інформаційний принцип : все інформаційне наповнення бази даних представлено одним і тільки одним способом, а саме - явним завданням значень атрибутів у кортежі відносин; зокрема, немає ніяких покажчиків (адрес), що зв'язують одне значення з іншим; наявність реляційної алгебри дозволяє

реалізувати декларативне програмування і декларативне опис обмежень цілісності, на додаток до навігаційного (процедурним) програмування і процедурної перевірці умов.

Принципи реляційної моделі були сформульовані в 1969 - 1970 роках Е. Ф. Коддом (EF Codd). Ідеї Кодда були вперше публічно викладені в статті "A Relational Model of Data for Large Shared Data Banks" [1], що стала класичною. Суворе виклад теорії реляційних баз даних (реляційної моделі даних) в сучасному розумінні можна знайти в книзі К. Дж. Дейта. "CJ Date. An Introduction to Database Systems" ("Дейт, К. Дж. Введення в системи баз даних"). Найбільш відомими альтернативами реляційної моделі є ієрархічна модель, і мережева модель. Деякі системи, що використовують ці старі архітектури, використовуються досі. Крім того, можна згадати про об'єктно-орієнтованої моделі, на якій будуються так звані об'єктно-орієнтовані СУБД, хоча однозначної і загальноприйнятого визначення такої моделі немає. Як правило, розрізняють два рівні абстракції представлення даних у вигляді інформаційної та фізичної моделей. Користувач мало звертає увагу на організацію фізичного зберігання інформації - його цікавить логічне представлення даних. Інформаційна модель повинна відображати предметну ділянку в зрозумілих і звичних для користувача термінах. Змістовно база даних містить структуровану певним чином інформацію про факти, події, об’єкти та їх властивості і зв’язки між ними. Концепція

реляційної моделі бази даних запропонована Е.Ф.Коддом в 1970 році для розв’язання наступної задачі - забезпечити незалежність представлення та опису даних від прикладних програм. В основі цієї моделі лежать поняття відношення (relations), подане у вигляді таблиці з дотриманням деяких обмежувальних умов. Основні взаємопов’язані поняття фізичного, спеціального прикладного та математичного рівнів побудови реляційної бази даних та їх взаємовідношення показані в таблиці, в рядках якої знаходяться еквівалентні поняття Стовпці відношення називають атрибутами і їм присвоюють імена. Список імен атрибутів відношення називається схемою відношення. Таблиці не дозволяють проводити узгодження наступних трьох способів маніпулювання даними: впорядкування, групування по певних ознаках, доступ по дереву параметрів. Це пов’язано з тим, що в таблиці всі три способи маніпулювання жорстко закріплені: дані впорядковані по одному параметру і не впорядковані по іншому.

Відношення реляційних баз даних. Відношення реляційної бази даних діляться на два класи: об’єктні та зв’язні. Об’єктне відношення зберігає дані про об’єкти (екземпляри сутності). В об’єктному відношенні один (або декілька) з атрибутів однозначно ідентифікують об’єкт. Такий ключовий атрибут називається (одиничним чи множинним) ключем відношень або первинним атрибутом. Ключ, як правило, знаходиться у першому стовпці. Інші

атрибути функціонально залежать від даного ключа. Ключ може включати кілька атрибутів (складний ключ). В об’єктному відношенні атрибути не повинні дублюватися. Це основне обмеження в реляційній базі даних для збереження цілісності даних. Зв’язне відношення зберігає ключі двох чи більше об’єктних відношень, тобто по ключах встановлюються зв’язки між об’єктами відношень. Зв’язне відношення може мати і інші атрибути, які функціонально залежать від цього зв’язку. Ключі в зв’язних відношеннях називаються зовнішніми (сторонніми) ключами, оскільки вони є первинними ключами інших відношень.

Умови і обмеження, які накладаються на відношення реляційних баз даних на табличному рівні представлення, можна сформулювати наступним чином: ·

не може бути однакових первинних ключів, тобто всі рядки (записи) повинні бути унікальними; ·

всі рядки повинні мати однакову типову структуру; ·

імена стовпців в таблиці повинні бути різними, а значення стовпців повинні бути однотиповими; ·

значення стовпців повинні бути атомарними, тобто не можуть бути компонентами інших відношень; ·

повинна зберігатися цілісність для зовнішніх ключів; ·

порядок розміщення рядків у таблиці неістотний - він впливає лише на швидкість доступу до потрібного рядка.

Кожен запис таблиці складається із окремих полів (атрибутів), які діляться на три групи: ключові, вказівні та допоміжні. Ключовими є ті атрибути, якими однозначно ідентифікується даний запис. Двох записів з однаковими ключовими атрибутами бути не може. Вказівні атрибути виконують роль ключових атрибутів в інших базах даних, на які посилається даний запис. З їх допомогою можна отримати додаткову інформацію для заданого запису. Допоміжні атрибути - це характеристики для заданого запису і вони, як правило, можуть повторюватись.

Файл бази даних може бути індексованим або ні. Наявність індексації означає, що всі записи в файлі баз даних перевпорядковані у відповідності із наперед заданим принципом (алфавітний порядок, порядок зростання чи спадання і так далі) по одному з полів. Інформація про індексацію міститься в індексному файлі у вигляді набору взаємопов’язаних пар чисел. Система індексації застосовується для зручності використання бази даних: для заданих полів створюється впорядкований перелік пар чисел, перше з яких вказує фізичний реальний порядковий номер запису в файлі, друге - порядковий номер даного запису у перевпорядкованому списку.

Розрізняють так звані одиничні і множинні індексні файли. Для кожного фізичного запису одиничний індексний файл містить тільки одну пару чисел, в той час як множинний - множину з декількох впорядкованих пар чисел. Кожне друге з наступної пари чисел в наборі означає порядковий номер у новому списку.

5.3. Архітектура «озеро даних» (Data Lake) для промисловості.

Концепт Data Lake (озеро даних) належить до архітектури та підходу до зберігання та опрацювання великих обсягів даних. Він пропонує централізоване зберігання різноманітних даних у неструктурованому або напівструктурованому вигляді.

Основна ідея Data Lake полягає у тому, щоб зібрати всі можливі джерела даних, такі як бази даних, журнали, файлові системи, сенсори, соціальні медіа тощо, у єдиному місці без необхідності стандартизації або опрацювання даних заздалегідь. Замість цього дані зберігають в їх первинному форматі, а також всю супутню контекстну інформацію.

Data Lake використовує розподілені системи зберігання, такі як Hadoop Distributed File System (HDFS) або cloud-платформи, для забезпечення масштабованості та надійності зберігання даних, що дає змогу використовувати паралельне опрацювання та аналіз для розуміння даних та отримання відповідних результатів.

Мета статті є аналіз концепту “озеро даних” та архітектур побудови інформаційних систем з його використанням.

Завдання дослідження – опрацювання варіантів формування рівня метаданих та конструктивів формування сховищ даних, джерелом для яких слугують озера даних, розгляд сфер застосування архітектур озер даних для побудови інформаційних систем освітньої галузі.

Поняття озера даних. Озера даних – доволі недосліджена сфера для багатьох рішень аналізу даних, вони стають все популярнішими в дослідженнях. Їх часто пов’язують із випадками опрацювання великих даних, використовують, наприклад, як центральні системи керування даними дослідницьких установ або як основну сутність процесів машинного навчання. Основна ідея збереження даних у їх рідному форматі в озері даних полегшує широкий спектр процедур опрацювання та поліпшує повторне використання даних. Однак зберігання таких величезних обсягів неопрацьованих даних створює певні проблеми, починаючи від загального моделювання даних та індексування для стислих запитів до інтеграції відповідних і масштабованих обчислювальних можливостей. Озеро розглядається насамперед як певний репозиторій для подальшого аналізу даних. Відповідно для отримання потрібних даних використовують алгоритми кластеризації, розпізнавання сутностей та зв’язків для кращого

розуміння схематичності озера даних. Модель роботи із сутностями подано на рис. 1.

Далі буде розглянуто спосіб подання озер за допомогою семантичних мереж із застосуванням багатошарових графів та навігаційного графа-моделі, що сприяють роботі із розпізнавання та зіставлення даних.

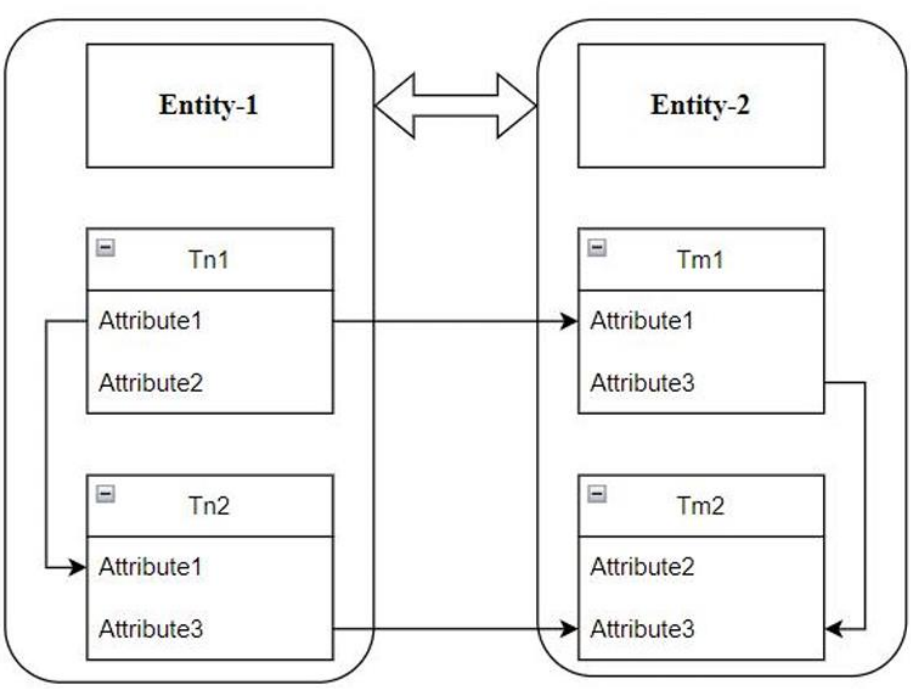


Рис. 1. Модель процесу розпізнавання сутностей

Архітектура озера даних описує структуру та компоненти системи, що вказує, як зберігати, упорядковувати та використовувати дані. Існує декілька варіацій архітектури озера даних:

• Архітектура резервуару розподіляє прийняті дані за їхнім статусом і використанням. Спочатку дані зберігаються у резервуарі неопрацьованих даних, потім перетворюються та переміщуються до відповідних резервуарів за типами даних. Крім того, відповідні

процеси застосовуються для підготовки даних для подальшого аналітичного опрацювання. Наприклад, аналогові дані, створені автоматизованим пристроєм, переміщуються до резервуара аналогових даних. Потім обсяг аналогових даних регулюється до можливого розміру (зменшення даних). Ставок архівних даних зберігає невикористані дані.

• Архітектура зон, яка розділяє етапи опрацювання кожного набору даних на різні етапи. Наприклад, можуть бути окремі зони для завантаження даних і перевірки їх якості, зберігання неопрацьованих даних, зберігання очищених і підтверджених даних, виявлення та вивчення даних або використання даних для аналізу.

• Загальна архітектура містить чотири рівні: приймання даних, зберігання, опрацювання та доступ. Запропоновано модель, яка забезпечує трирівневий функціонально-орієнтований зв’язок, схематично зображену на рис. 2.

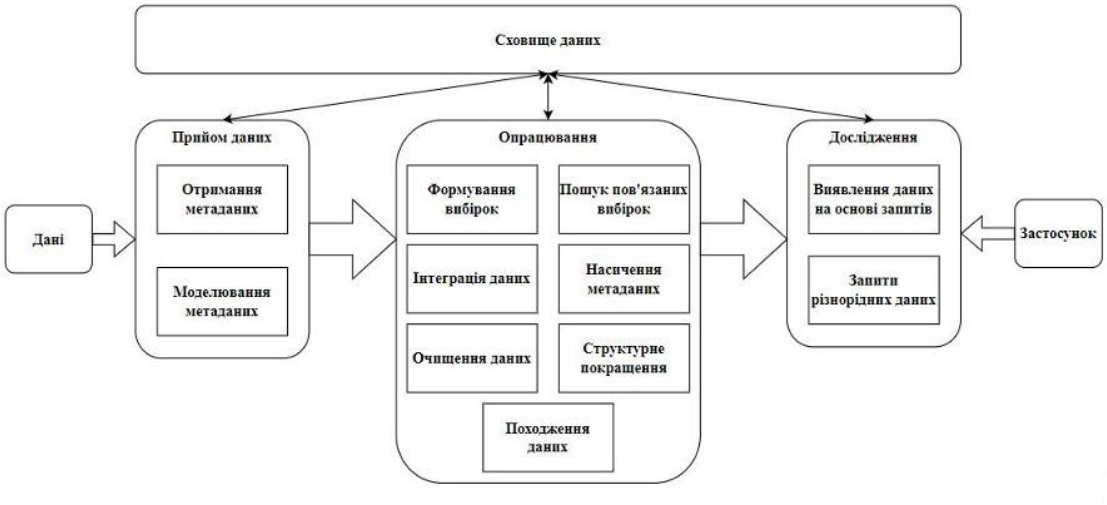


Рис. 2. Модель архітектури трирівневого функціонально-орієнтованого зв'язку

Оскільки озера даних постійно наповнюються, то у них можуть з’являтись і нові залежності між даними, які раніше не було можливості пов’язати. Також за рахунок великої кількості джерел з’являтимуться суперечливі або не зовсім однозначні дані щодо певного об’єкта, які не можна просто так об’єднати чи надати перевагу тільки частковим показникам. У такому випадку пропонують використовувати онтології та семантичні правила, для вирішення проблем неоднорідності й чіткого розуміння тих чи інших даних.

Робота з даними. Модель озер даних зосереджена на тому, що потрібно зберігати інформацію, а не на тому, які власне дані потрібні чи яке їх призначення. Тому можна доповнити озера семантичною інформацією, яка б формувала правила підбору даних. Для цього використовують семантичні мережі із застосуванням багатошарових графів.

Ще один варіант роботи з графами для кращого розуміння структури озера даних – створення навігаційного графа-моделі озера. Кожен вузол поданий набором атрибутів і ребра, що будуть зв'язками між вузлами. Відповідно до результатів цього дослідження, користувачі отримували результати, які неможливо одержати через пошук за ключовими словами.

Серед інших можливостей роботи із даними – опрацювання великої кількості документації, різних видів та типів. Для аналізу і управління такими даними пропонують застосовувати Large Language Models(LLM), які попередньо навчаються на широкому спектрі даних.

Стосовно опрацювання наборів даних науковці пропонують виявлення особливих наборів зі спільною семантикою. Пошук таких вибірок здійснюється через зіставлення контексту та об'єднання таблиць, що можуть бути пов'язані.

Нині найпрактичнішою у плані архітектури озера даних є зонна. Але науковці шукають альтернативи та пропонують модифіковані варіанти, наприклад архітектура на основі FAIR Digital Objects(FDO). Такий підхід до архітектури ґрунтується на понятті гнучкості озера даних і на більш прямому зв'язку користувача і системи. Якщо користувач бажає опрацювати певні дані, він повинен спочатку налаштувати типи, новий тип метаданих, який міститиме пари ключ – значення відповідних відображень. Далі

озеро приймає ці дані й “оцифровує” відповідно до заданого маніфесту – набору семантичних правил.

Використання озер даних пов’язують із багатьма складнощами, основні з них: складність (проблема перетворення даних), вартість обчислень (обчислення даних великої розмірності доволі вартісне), різноманітність (пов’язана із нечіткістю управління даними), інтеграція у пам’яті (виведення даних, із кількох джерел, без їх об’єднання), відсутність чітких рішень (недостатній рівень контролю даних та великих даних), проблема узагальнення (аналіз стосується здебільшого специфічної предметної області), масштабованість (постійний приплив нових даних) та варіативність (неструктурованість та неоднорідність даних).

Під час роботи з озерами даних може виникати проблема із джерелами даних, власне із їх достовірністю або змінами домену (особливо якщо це стосується постійно повторюваної вхідної інформації). На такий випадок пропонують функції автоматичної неконтрольованої перевірки джерел.

Результати досліджень із використанням озер даних.У мережі інтернет наявні джерела, які стосуються досліджень, пов’язаних зі створенням структур для запису даних та метаданих у випадку опрацювання векторних і растрових просторових даних, зокрема даних просторового розташування та їх відповідного відображення.

Ще однією сферою застосування озер даних є гуманітарне спрямування, зокрема археологічні проєкти. Основна проблема археологічних даних – їх різноманітність, зокрема текстові документи, зображення, дані давачів, документи, звіти про розкопки, створюються різними програмними чи апаратними засобами, які не завжди взаємосумісні. Схожий також фреймворк PEXESO, орієнтований на роботу із виявлення сумісних наборів даних. Дослідники звертають увагу на проблеми, які можуть виникати за великих розмірностей, а також у разі використання різних форматів даних та помилки у наповненні текстових даних.

Платформа Jupyter Notebook використовує озера даних для покращення обміну даними між науковцями через компонування пов’язаних наборів даних. Це стосується навчальних та дослідницьких даних, серед особливо важливих – функції зменшення розмірностей та роботи з метаданими. Серед найчастіше використовуваних типів файлів зберігання даних – тип Parquet, але його недолік – неможливість працювати із геопросторовими даними. Порівняно недавно створено розширення Spatial Parquet, яке цілком стабільно працює зі стандартним Parquet, та створює додаткові функції для поліпшення аналізу.

Для покращення роботи користувача науковці досліджують роботу із запитами до озер даних природною мовою. Дослідники

пропонують проєктне рішення SympHony, яка має розкладати запити природною мовою на підзапити, оцінювати їх певними наборами даних і видавати результат об'єднанням цих підзапитів.

Цікаві дослідження, які пропонують застосування озер даних у галузі страхування. Хоч таке рішення супроводжується додатковими ризиками, його все одно вважають дуже перспективним. Для страхових компаній можливість отримувати інформацію про клієнтів буде безцінною. Аналогічно специфічним є застосування цього концепту в медичних структурах, оскільки медичні дані часто оберігаються лікарською таємницею і їх можна надавати для оприлюднення чи аналізу тільки за певними юридичними аспектами.

Контрольні запитання.

1. Які два основні типи баз даних із точки зору організації даних?
2. Яка ключова відмінність між реляційними та нереляційними базами даних?
3. Назвіть чотири основні види нереляційних (NoSQL) баз даних та їхнє основне призначення.
4. Для яких завдань краще підходить графова база даних?

Лекція 6: Вибірка та аналіз даних з баз

6.1. Мова SQL для роботи з БД: Основні оператори: SELECT, FROM, WHERE.

Основні поняття. Всі мови маніпулювання даними (ММД), створені до появи реляційних баз даних і розроблені для багатьох систем управління базами даних (СКБД) персональних комп'ютерів, були орієнтовані на операції з даними, представленими у вигляді логічних записів файлів. Це вимагало від користувачів детального знання організації зберігання даних і достатніх зусиль для вказівки не лише того, які дані потрібні, але й того, де вони розміщені і як крок за кроком отримати їх.

Розглянутий ж нижче непроцедурного мова SQL (Structured Query Language - структуризовано мова запитів) орієнтований на операції з даними, представленими у вигляді логічно взаємозалежних сукупностей таблиць. Особливість пропозицій цієї мови полягає в тому, що вони орієнтовані більшою мірою на кінцевий результат обробки даних, ніж на процедуру цієї обробки. SQL сам визначає, де знаходяться дані, які індекси і навіть найбільш ефективні послідовності операцій слід використовувати для їх отримання: не треба вказувати ці деталі в запиті до бази даних.

Для ілюстрації відмінностей між ММД розглянемо таку ситуацію. Нехай, наприклад, ви збираєтеся подивитися кінофільм і хочете скористатися для поїздки в кінотеатр послугами таксі. Одному шоферу таксі достатньо сказати назву фільму - і він сам знайде вам кінотеатр, в якому показують потрібний фільм. (Подібним же чином, самостійно, відшукує запитані дані SQL.)

Для іншого шофера таксі вам, можливо, потрібно буде самому дізнатися, де демонструється потрібний фільм і назвати кінотеатр. Тоді водій повинен знайти адресу цього кінотеатру. Може статися й так, що вам доведеться самому дізнатися адресу кінотеатру і запропонувати водієві проїхати до нього по таким-то і таким-то вулицях. У найгіршому випадку вам, може бути, навіть доведеться по дорозі давати вказівки: "Повернути наліво ... проїхати п'ять кварталів ... повернути праворуч ...". (Аналогічно більший або менший рівень деталізації запиту доводиться створювати користувачеві в різних СУБД, що не мають мови SQL.) Поява теорії реляційних баз даних і запропонованого Коддом мови запитів "alpha", заснованого на реляційному обчисленні [2, 3], ініціювало розробку ряду мов запитів, які можна віднести до двох класів:

Алгебраїчні мови, що дозволяють виражати запити засобами спеціалізованих операторів, що застосовуються до відносинам (JOIN - з'єднати, INTERSECT - перетнути, SUBTRACT - відняти і т.д.).

Мови числення предикатів представляють собою набір правил для запису виразу, що визначає нове відношення з заданої сукупності існуючих відносин. Іншими словами числення предикатів є метод визначення того ставлення, яке нам бажано отримати (як відповідь на запрос) з відносин, вже наявних в базі даних.

Розробка, в основному, йшла у відділеннях фірми IBM (мови ISBL, SQL, QBE) і університетах США (PIQUE, QUEL) [3]. Останній створювався для СУБД INGRES (Interactive Graphics and Retrieval System), яка була розроблена на початку 70-х років в Університеті шт. Каліфорнія і сьогодні входить до п'ятірки кращих професійних СУБД. Сьогодні з усіх цих мов повністю збереглися і розвиваються QBE (Query-By-Example - запит по зразку) та SQL, а з решти взяті в розширення внутрішніх мов СУБД тільки найцікавіші конструкції.

На початку 80-х років SQL "переміг" інші мови запитів і став фактичним стандартом таких мов для професійних реляційних СУБД. У 1987 році він став міжнародним стандартом мови баз даних і почав впроваджуватися у всі поширенням СУБД персональних комп'ютерів. Чому ж це сталося? Безперервне зростання швидкодії, а також зниження енергоспоживання, розмірів і вартості комп'ютерів привели до різкого розширення можливих ринків їх збуту, кола користувачів, різноманітності типів і цін. Природно, що розширився попит на різноманітне програмне забезпечення. Борючись за

покупця, фірми, що виробляють програмне забезпечення, стали випускати на ринок все більш і більш інтелектуальні і, отже, об'ємні програмні комплекси. Купуючи (бажаючи придбати) такі комплекси, багато організацій та окремі користувачі часто не могли розмістити їх на власних ЕОМ, однак не хотіли і відмовлятися від нового сервісу. Для обміну інформацією та її усуспільнення були створені мережі ЕОМ, де усуспільнюватись програми і дані стали розміщувати на спеціальних обслуговуючих пристроях - файлових серверах.

СУБД, що працюють з файловими серверами, дозволяють безлічі користувачів різних ЕОМ (іноді розташованих досить далеко один від одного) отримувати доступ до одних і тих же баз даних. При цьому спрощується розробка різних автоматизованих систем управління організаціями, навчальних комплексів, інформаційних та інших систем, де безліч співробітників (учнів) повинні використовувати загальні дані й обмінюватися створюваними в процесі роботи (навчання). Однак при такій ідеології вся обробка запитів з програм або з терміналів користувача ЕОМ виконується на цих же ЕОМ. Тому для реалізації навіть простого запиту ЕОМ часто повинна зчитувати з файлового сервера і (або) записувати на сервер цілі файли, що веде до конфліктних ситуацій і перевантаження мережі. Для виключення зазначених та деяких інших недоліків була запропонована технологія "Клієнт-Сервер", по якій запити

користувальницьких ЕОМ (Клієнт) обробляються на спеціальних серверах баз даних (Сервер), а на ЕОМ повертаються лише результати обробки запиту. При цьому, природно, потрібен єдину мову спілкування з Сервером і в якості такої мови обраний SQL. Тому всі сучасні версії професійних реляційних СУБД (DB2, Oracle, Ingres, Informix, Sybase, Progress, Rdb) і навіть нереляційних СУБД (наприклад, Adabas) використовують технологію "Клієнт-Сервер" і мову SQL. До того ж приходять розробники СУБД персональних ЕОМ, багато з яких вже сьогодні забезпечені мовою SQL. Існує думка: Оскільки велика частина запитів формулюється на SQL, практично байдуже, що це за СУБД - був би SQL.

Реалізація в SQL концепції операцій, орієнтованих на табличне представлення даних, дозволило створити компактний мову з невеликим (менше 30) набором пропозицій. SQL може використовуватися як інтерактивний (для виконання запитів) і як вбудований (для побудови прикладних програм).

У ньому існують:

пропозиції визначення даних (визначення баз даних, а також визначення та знищення таблиць і індексів);

запити на вибір даних (пропозиція SELECT);

пропозиції модифікації даних (додавання, видалення і зміна даних);

пропозиції керування даними (надання та скасування привілеїв на доступ до даних, управління транзакціями і інші).

Крім того, він надає можливість виконувати в цих пропозиціях: арифметичні обчислення (включаючи різноманітні функціональні перетворення), обробку текстових рядків і виконання операцій порівняння значень арифметичних виразів і текстів; упорядкування рядків і (або) стовпців при виведенні вмісту таблиць на друк або екран дисплея; створення уявлень (віртуальних таблиць), що дозволяють користувачам мати свій погляд на дані без збільшення їх обсягу в базі даних; запам'ятовування виведеного за запитом вмісту таблиці, декількох таблиць або уявлення в іншій таблиці (реляційна операція присвоювання). агрегатування даних: групування даних і застосування до цих груп таких операцій, як середнє, сума, максимум, мінімум, число елементів і т.п.

У SQL використовуються наступні основні типи даних, формати яких можуть дещо відрізнятися для різних СУБД:

INTEGER - Ціле число (зазвичай до 10 значущих цифр і знак);

SMALLINT - "Коротке ціле" (зазвичай до 5 значущих цифр і знак);

DECIMAL (p, q) - Десяткове число, яке має p цифр (0 <p <16) і знак;

за допомогою q задається число цифр праворуч від десяткової точки (q <p, якщо q = 0, воно може бути опущено);

FLOAT - Дійсне число з 15 значущими цифрами і цілочисловим порядком, визначеним типом СУБД;

CHAR (n) - Символьний рядок фіксованої довжини з n символів (0 <n <256);

VARCHAR (n) - Символьний рядок змінної довжини, що не перевищує n символів (n> 0 і різне в різних СУБД, але не менше 4096);

DATE - Дата в форматі, визначеному спеціальною командою (за замовчуванням mm / dd / yy); поля дати можуть містити тільки реальні дати, що починаються за кілька тисячоліть до н.е. і обмежені п'ятим-десяти тисячоліть н.е.;

TIME - Час у форматі, визначеному спеціальною командою, (за замовчуванням hh.mm.ss);

DATETIME - Комбінація дати і часу;

MONEY - Гроші в форматі, який визначає символ грошової одиниці ($, грн, ...) і його розташування (суфікс чи префікс), точність дробової частини і умова для показу грошового значення.

У деяких СУБД ще існує тип даних LOGICAL, DOUBLE та ряд інших. СУБД INGRES надає користувачеві можливість самостійного визначення нових типів даних, наприклад, площинні або просторові координати, одиниці різних метрик, п'яти-або шестиденні тижня (робочий тиждень, де відразу після п'ятниці або суботи слід понеділок), дробу, графіка, великі цілі числа (що стало дуже актуальним для російських банків) і т.п.

Орієнтований на роботу з таблицями SQL не має достатніх коштів для створення складних прикладних програм. Тому в різних СУБД він або використовується разом з мовами програмування високого рівня (наприклад, такими як Сі чи Паскаль), або включений до складу команд спеціально розробленої мови СУБД (язик систем dBASE, R: BASE і т.п.).

Уніфікація повних мов сучасних професійних СУБД досягається за рахунок впровадження об'єктно-орієнтованої мови четвертого покоління 4GL. Останній дозволяє організовувати цикли, умовні речення, меню, екранні форми, складні запити до баз даних з інтерфейсом, орієнтованим як на алфавітно-цифрові термінали, так і на віконний графічний інтерфейс (XWindows, MS-Windows).

Запити на читання даних. Всі запити на отримання практично будь-якої кількості даних з однієї або декількох таблиць виконуються за допомогою єдиного пропозиції SELECT. У

загальному випадку результатом реалізації пропозиції SELECT є інша таблиця (див. приклади п. 1.3). До цієї нової (робочої) таблиці може бути знову застосована операція SELECT і т.д., тобто такі операції можуть бути вкладені одна в одну. Представляє історичний інтерес той факт, що саме можливість включення одного речення SELECT всередину іншого послужила мотивуванням використання прикметника "структуризувати" в назві мови SQL.

Пропозиція SELECT може використовуватися як:

самостійна команда на отримання і вивід рядків таблиці, сформованої із стовпців і рядків однієї або декількох таблиць (подань);

елемент WHERE-або HAVING-умови (скорочений варіант пропозиції, званий "вкладений запит");

фраза вибору в командах CREAT VIEW, DECLARE CURSOR або INSERT;

засіб привласнення глобальним змінним значень із рядків сформованої таблиці (INTOфраза).

У даній і наступній главах будуть розглянуті тільки два перші функції пропозиції SELECT, а тут - його синтаксис, обмежений конструкціями, що використовуються при реалізації цих функцій.

Тут (так само як і в інших розділах книги) в синтаксичних конструкціях використовуються такі позначення:

зірочка (*) для позначення "все" - вживається в звичайному для програмування сенсі, тобто "Всі випадки, задовольняють визначенню";

квадратні дужки ([]) - означають, що конструкції, укладені в ці дужки, є необов'язковими (тобто можуть бути опущені);

фігурні дужки ({}) - означають, що конструкції, укладені в ці дужки, повинні розглядатися як цілі синтаксичні одиниці, тобто вони дозволяють уточнити порядок розбору синтаксичних конструкцій, замінюючи звичайні дужки, які використовуються в синтаксисі SQL;

три крапки (...) - вказує на те, що безпосередньо передує йому синтаксична одиниця факультативно може повторюватися один або більше разів;

пряма риса (|) - означає наявність вибору з двох або більше можливостей.

Наприклад позначення ASC | DESC вказує, можна вибрати один з термінів ASC або DESC; коли ж один із елементів вибору укладений у квадратні дужки, то це означає, що він вибирається за

замовчуванням (так, [ASC] | DESC означає, що відсутність усієї цієї конструкції буде сприйматися як вибір ASC);

крапка з комою (;) - завершальний елемент пропозицій SQL;

кома (,) - використовується для розділення елементів списків;

прогалини () - можуть вводитися для підвищення наочності між будь-якими синтаксичними конструкціями пропозицій SQL;

прописні жирні латинські літери та символи - використовуються для написання конструкцій мови SQL і повинні (якщо це спеціально не обумовлено) записуватися в точності так, як показано;

малі літери - використовуються для написання конструкцій, які повинні замінюватися конкретними значеннями, обраними користувачем, причому для визначеності окремі слова цих конструкцій зв'язуються між собою символом підкреслення (_);

терміни таблиця, стовпець, ... - Замінюють (з метою скорочення тексту синтаксичних конструкцій) терміни ім'я_таблиці, ім'я_стовпця, ..., відповідно; термін таблиця - використовується для узагальнення таких видів таблиць, як базовая_табліца, уявлення або псевдонім; тут псевдонім служить для тимчасового (на момент виконання запиту) перейменування та (або) створення робочої копії базовой_табліци (подання).

Пропозиція SELECT (вибрати) має наступний формат: підзапит [UNION [ALL] підзапит] ...

[ORDER BY {[таблиця.] Стовпець | номер_елемента_SELECT} [[ASC] | DESC] [, {[Таблиця.] Стовпець | номер_елемента_SELECT} [[ASC] | DESC]] ...; і дозволяє об'єднати (UNION) а потім упорядкувати (ORDER BY) результати вибору даних, отриманих за допомогою декількох "підзапитів". При цьому упорядкування можна проводити в порядку зростання - ASC (ASCending) або убування DESC (DESCending), а за замовчуванням приймається ASC.

У цьому реченні підзапит дозволяє вказати умови для вибору потрібних даних і (якщо потрібно) їх обробки

SELECT (Вибрати) дані із зазначених стовпців і (якщо необхідно) виконати перед виведенням їх перетворення відповідно до зазначеними виразами і (або) функціями

FROM (З) перерахованих таблиць, в яких розташовані ці стовпці

WHERE (Де) рядки із зазначених таблиць повинні задовольняти зазначеного переліку умов відбору рядків

GROUP BY (Групуючи по) зазначеного переліку стовпців з тим, щоб отримати для кожної групи єдине агреговане значення, використовуючи у фразі SELECT SQL-функції SUM (сума), COUNT

(кількість), MIN (мінімальне значення), MAX (максимальне значення) або AVG (середнє значення)

HAVING (Маючи) в результаті лише ті групи, які задовольняють зазначеному переліку умов відбору груп і має формат

SELECT [[ALL] | DISTINCT] {* | елемент_SELECT [, елемент_SELECT] ...}

FROM {базовая_табліца | уявлення} [псевдонім] [, {Базовая_табліца | уявлення} [псевдонім]] ...

[WHERE фраза]

[GROUP BY фраза [HAVING фраза]];

Елемент_SELECT - це одна з наступних конструкцій: [Таблиця.] * | Значення | SQL_функція | сістемная_переменная

де значення - це:

[Таблиця.] Стовпець | (вираз) | константа | змінна

Синтаксис виразів має вигляд ({[[+] | -] {Значення | функція_СУБД} [+ | - | * | **]} ...)

а синтаксис SQL_функцій - одна з наступних конструкцій: {SUM | AVG | MIN | MAX | COUNT} ([[ALL] | DISTINCT] [таблиця.]

Стовпець) {SUM | AVG | MIN | MAX | COUNT} ([ALL] вираз) COUNT (*)

Фраза WHERE включає набір умов для відбору рядків:

WHERE [NOT] WHERE_условіе [[AND | OR] [NOT] WHERE_умова] ..

де WHERE_ умова - одна з наступних конструкцій:

значення {= | <> | <| <= |> |> =} {значення | (підзапит)}

значення_1 [NOT] BETWEEN значення_2 AND значеніе_3

значення [NOT] IN {(константа [, константа] ...) | (підзапит)}

значення IS [NOT] NULL

[Таблиця.] Стовпець [NOT] LIKE 'строка_сімволов' [ESCAPE 'символ']

EXISTS (підзапит). Крім традиційних операторів порівняння (= | <> | <| <= |> |> =) в WHERE фразі використовуються умови BETWEEN (між), LIKE (схоже на), IN (належить), IS NULL (не визначено) і EXISTS (існує), які можуть передувати оператором NOT (не). Критерій відбору рядків формується з одного або декількох умов, з'єднаних логічними операторами:

AND - Коли повинні задовольнятися обидва поділюваних за допомогою AND умови; OR - Коли має задовольнятися одне з поділюваних за допомогою OR умов;

AND NOT - Коли має задовольнятися перша умова і не повинно друге;

OR NOT - Коли або має задовольнятися перша умова або не має задовольнятися друге, причому існує пріоритет AND над OR (спочатку виконуються всі операції AND і тільки після цього операції OR). Для отримання бажаного результату WHERE умови мають бути введені в правильному порядку, який можна організувати введенням дужок. При обробці умови числа порівнюються алгебраїчно - негативні числа вважаються меншими, ніж позитивні, незалежно від їх абсолютної величини. Рядки символів порівнюються відповідно до їх поданням в коді, використовуваному в конкретної СУБД, наприклад, в коді ASCII. Якщо порівнюються два рядки символів, що мають різні довжини, більш короткий рядок доповнюється праворуч пробілами для того, щоб вони мали однакову довжину перед здійсненням порівняння.

Нарешті, синтаксис фрази GROUP BY має вигляд GROUP BY [таблиця.] Стовпець [, [таблиця.] Стовпець] ... [HAVING фраза]

GROUP BY ініціює перекомпоновку формованої таблиці по групах, кожна з яких має однакове значення в стовп-цях, включених до переліку GROUP BY. Далі до цих груп застосовуються агрегуються функції, зазначені у фразі SELECT, що призводить до заміни всіх значень групи на єдине значення (сума, кількість тощо). За допомогою фрази HAVING (синтаксис якої майже не відрізняється від синтаксису фрази WHERE)

HAVING [NOT] HAVING_условіе [[AND | OR] [NOT] HAVING_условіе] ... можна виключити з результату групи, не задовольняють заданим умовам: значення {= | <> | <| <= |> |> =} {значення | (підзапит) | SQL_функція}

{Значення_1 | SQL_функція_1} [NOT] BETWEEN {Значення_2 | SQL_функція_2} AND {значеніе_3 | SQL_функція_3}

{Значення | SQL_функція} [NOT] IN {(константа [, константа] ...) | (Підзапит)}

{Значення | SQL_функція} IS [NOT] NULL

[Таблиця.] Стовпець [NOT] LIKE 'строка_сімволов' [ESCAPE 'символ']

EXISTS (підзапит).

Агрегування даних.

У SQL існує низка спеціальних стандартних функцій (SQL-функцій). Крім спеціального випадку COUNT (*) кожна з цих функцій оперує сукупністю значень стовпця деякої таблиці і створює єдине значення, яке визначається так:

COUNT - Число значень в стовпці,

SUM - Сума значень в стовпці,

AVG - Середнє значення в стовпці,

MAX - Найбільше значення в стовпці,

MIN - Найменше значення в стовпці.

Для функцій SUM і AVG розглянутий стовпець повинен містити числові значення. Слід зазначити, що тут стовпець - це стовпець віртуальної **таблиці, в якій можуть міститися дані не тільки з шпальти базової таблиці, але й дані, отримані шляхом функціонального перетворення і (або) зв'язування символами арифметичних операцій значень з одного або декількох стовпців**. При цьому вираз, що визначає стовпець такої таблиці, може бути як завгодно складним, але не повинно містити SQL-функцій (вкладеність SQL-функцій не допускається). Однак з SQL-функцій можна складати будь-які вирази.

Аргументу всіх функцій, крім COUNT (*), може передувати ключове слово DISTINCT (різний), що вказує, що надлишкові дублюючі значення повинні бути виключені перед тим, як буде застосовуватися функція. Спеціальна ж функція COUNT (*) служить для підрахунку всіх без винятку рядків у таблиці (включаючи дублікати).

- **6.2. З'єднання таблиць (JOIN) для отримання комплексної інформації.**

JOIN — це операція в SQL, яка дозволяє поєднувати дані з двох або більше таблиць на основі логічного зв'язку між ними.

Основна мета: Отримати об'єднану інформацію, яка розподілена між різними таблицями, для комплексного аналізу.

Розглянемо на прикладі двох таблиць:

Таблиця employees (співробітники):

id	name	department_id
1	Іван	1
2	Марія	2
3	Петро	1
4	Олена	NULL

Таблиця departments (відділи):

id	name
1	IT
2	Маркетинг
3	Бухгалтерія

INNER JOIN

Повертає тільки ті рядки, де є збіг в обох таблицях

SELECT e.name, d.name as department

FROM employees e

INNER JOIN departments d ON e.department_id = d.id;

Результат:

name	department
Іван	IT
Марія	Маркетинг
Петро	IT

LEFT JOIN

Повертає всі рядки з лівої таблиці + відповідні з правої

SELECT e.name, d.name as department

FROM employees e

LEFT JOIN departments d ON e.department_id = d.id;

name	department
Іван	IT
Марія	Маркетинг
Петро	IT
Олена	NULL

RIGHT JOIN

Повертає всі рядки з правої таблиці + відповідні з лівої

SELECT e.name, d.name as department

FROM employees e

RIGHT JOIN departments d ON e.department_id = d.id;

Результат:

name	department
Іван	IT
Марія	Маркетинг
Петро	IT
NULL	Бухгалтерія

FULL OUTER JOIN

Повертає всі рядки з обох таблиць

```
SELECT e.name, d.name as department
FROM employees e
FULL OUTER JOIN departments d ON e.department_id = d.id;
```

Результат:

name	department
Іван	IT
Марія	Маркетинг
Петро	IT
Олена	NULL
NULL	Бухгалтерія

Ключові моменти для ефективного використання JOIN:

1. Завжди використовуйте ALIAS (псевдоніми) для таблиць
2. Вказуйте конкретні стовпці замість `SELECT *`
3. Фільтруйте дані за допомогою `WHERE` після JOIN
4. Використовуйте індекси на полях, за якими відбувається з'єднання
5. Уважно вибирайте тип JOIN залежно від потрібного результат

Контрольні запитання.

1. Що таке база даних (БД)?

2. Назвіть основні об'єкти реляційної бази даних.

3. Для чого використовується SQL?

4. Яка різниця між командами `WHERE` та `HAVING` в SQL?

Лекція 7: Методи обробки зображень у комп’ютерному зорі

7.1. Характеристики якості зображення.

У фахівців не існує єдиної думки щодо загального визначення поняття "зображення" [1]. Надалі під терміном "зображення" будемо розуміти візуальне зображення, а саме все те, що можна побачити оком чи зареєструвати системою штучного зору. Зображення можна представити різними способами, які варто поділити на неформалізовані та формалізовані. Неформалізованими є способи представлення зображень фотографіями, слайдами, фільмами, картинами, малюнками й іншими фізичними носіями розподілу яскравості та кольору, зокрема, штриховим чи растровим друком на папері, на екрані і т.ін. Дані способи подають зображення природно, неформально і призначені для безпосереднього пред'явлення зображень людині чи системі штучного зору. Вони не дозволяють проводити розробку й аналіз систем обробки зображень і тому далі розглядатися не будуть. Формалізовані способи представлення зображень мають за мету одержання певних математичних моделей зображень, які можна піддавати аналізу й вивчати. Одним із способів формалізованого представлення є спосіб математичного опису функціональною залежністю. Він полягає в заданні однозначної функції, яка кожному значенню свого аргументу зіставляє одне певне значення; самі значення аргументів та функції можуть виражатися цілими або дійсними числами, впорядкованими парами

таких чисел, трійками тощо. Якщо область визначення й область значень даної функції є скінченними множинами, то таку функцію можна задати способом переліку, вказуючи впорядковані пари (аргументи, значення), що приводить до формалізованого представлення зображення у вигляді таблиці (таблиць) чисел. Цей спосіб є зручним для представлення цифрових зображень [2]. Інші способи формалізованого представлення зображень випливають з певних математичних теорій і зазвичай призначені для рішення окремих задач обробки зображень. Так, для рішення задач аналізу зображень може застосовуватися символічний опис зображень [3]; для рішення задач аналізу сцен – опис зображень у термінах теорії формальних граматик [3]; для рішення задач обробки бінарних зображень – опис зображень як множин, що задані у векторному просторі [4], тощо. Надалі будемо розглядати тільки спосіб опису зображень функціональною залежністю як найбільш загальний спосіб формалізованого представлення зображень.

Будь-яке зображення можна описати векторною функцією в певному N-вимірному лінійному просторі [5]. Компонентами даної векторної функції звичайно є тривимірні скалярні дійсні функції просторових координат x, y та часу t. Наприклад, для зображень, отриманих за допомогою спектрозональних систем, розмірність N лінійного простору дорівнює числу спектральних зон [2]. Такі спектрозональні зображення описують векторною функцією виду:

$$\vec{f}(x,y,t) = f_1(x,y,t)\,\vec{i}_1 + f_2(x,y,t)\,\vec{i}_2 + \ldots + f_N(x,y,t)\,\vec{i}_N \quad , \tag{1.1}$$

Де $\vec{i}_1, \vec{i}_2, \ldots, \vec{i}_N$ – орти (одиничні вектори) N-вимірного лінійного простору, що визначають спектральні зони зображення; $f_i(x,y,t); \quad i = 1,\ldots,N$ – тривимірні дійсні функції аргументів x, y, t. Формула (1.1) вказує, що будь-яке значення векторної функції $\vec{f}(x,y,t)$ являє собою впорядковану N-ку дійсних чисел.

Кольорове зображення описують тривимірною векторною функцією в обраній системі координат кольору, наприклад, у RGB-координатній системі:

$$\vec{f}(x,y,t) = f_R(x,y,t)\,\vec{r} + f_G(x,y,t)\,\vec{g} + f_B(x,y,t)\,\vec{b} \quad , \tag{1.2}$$

де $\vec{r}, \vec{g}, \vec{b}$ – орти тривимірного кольорового куба [3], що відповідають червоній (R – red), зеленій (G – green) і синій (B – blue) компонентам кольору. З (1.2) видно, що значення векторної функції $\vec{f}(x,y,t)$ залежать від аргументів x, y, t та належать тривимірному лінійному векторному простору, тобто являють собою впорядковані трійки дійсних чисел. Слід зазначити, що для опису кольорового зображення можуть використовуватися й інші кольорові координатні системи [3, 6], які при рішенні певних задач дозволяють досягти переваг у порівнянні з RGBкоординатною системою. Але на практиці найбільш часто використовується саме RGB-координатна система.

З представлень (1.1) і (1.2) випливає, що просторово-часова обробка спектрозональних і кольорових зображень зводиться до просторово-часової обробки їх кожної компоненти. Наприклад, просторово-часова обробка кольорового зображення може виконуватися за схемою, наведеною на рис. 1.1. Якщо просторово-часова обробка кожної компоненти не залежить від аналогічної обробки інших компонент, то вона може виконуватися паралельно іншим.

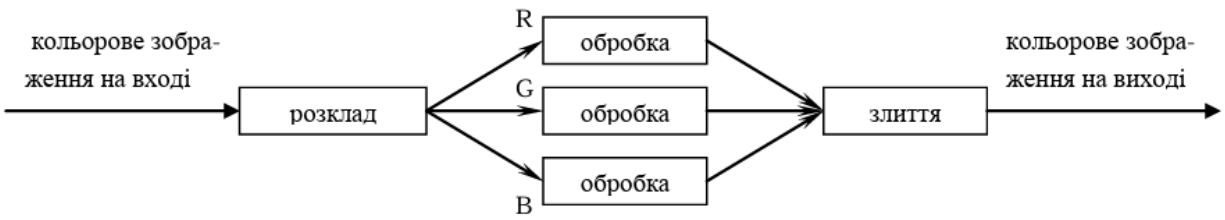


Рис. 1.1. Схема просторово-часової обробки кольорового зображення

Крім просторово-часової обробки, існують й інші види обробки спектрозональних і кольорових зображень, що засновані на спільній обробці двох чи більше компонент зображення. Але просторово-часова обробка є основним і найбільш важливим видом [1–7]. Для опису однокольорового (монохроматичного) зображення використовують лише скалярну функцію, покладаючи розмірність лінійного простору N=1.

Надалі обмежимося випадком статичних однокольорових зображень, видаляючи із розгляду залежність зображення від часу t.

Зображення, які описуються відповідними скалярними двовимірними функціями, будуть позначатися f(x,y), де f(x,y) – дійсна функція просторових координат x, y. Значення функції f(x,y) у кожній просторовій точці (x,y) називають яскравістю зображення чи тоном, а власне саме таке зображення – напівтоновим. Якщо функція f(x,y) може приймати лише два можливих значення, то відповідне їй зображення називають бінарним.

Якщо змінні x, y і значення функції f(x,y) змінюються неперервно, то відповідне зображення називають аналоговим, чи неперервним. Якщо змінні x, y є дискретними, а значення функції f(x,y) є неперервними, то відповідне зображення називають дискретним, причому замість f(x,y) використовують позначення f(m,n) чи fm,n, де m, n – цілі числа, що відповідають точкам просторової дискретизації на координатній площині xOy. Якщо дискретне зображення є дискретним і за яскравістю, причому яскравість може приймати значення тільки зі скінченної множини значень, то його називають цифровим. Перехід від аналогового зображення до дискретного зображення і назад може бути виконано без втрат (якщо виконуються умови теореми Котельникова – Шеннона про частоту взяття просторових відліків [2]). Але перехід від аналогового зображення до цифрового зображення і назад практично завжди пов'язаний із втратами через нелінійність операції дискретизації яскравості зображення. Ці втрати можна тільки

зменшити за рахунок збільшення кількості рівнів квантування яскравості.

Клас функцій, що описують неперервні напівтонові зображення, часто звужують. Так, звичайно вимагають, щоб функція f(x,y) задовольняла таким умовам [5]: 1) мала “гарну” поведінку (наприклад, була аналітичною функцією); 2) була обмеженою функцією, тобто $0 \le f(x,y) \le A < \infty$; для усіх x, y з області визначення функції. Якщо перша умова відбиває вимогу до "гладкості" зображення, то друга пов'язана з тим, що енергія джерела світлового випромінювання є пропорційною до квадрата амплітуди електричної компоненти електромагнітного поля і тому являє собою невід'ємну величину [2].

Існують різні класи задач просторової обробки напівтонових зображень [1–7]; серед них важливе місце займають задачі просторового аналізу та просторового синтезу зображень.

Задача просторового аналізу зображень полягає в одержанні опису зображення через набір заданих просторових ознак [2, 3]. Для цього досить часто використовують опис зображень через набір просторових гармонік. Для одержання набору просторових гармонік напівтонове зображення розкладають у ряд Фур'є [1–3].

Неперервне напівтонове зображення можна адекватно і без втрати інформації описати його просторовим спектром. Для отримання просторового спектра неперервного напівтонового

зображення використовують пряме двовимірне перетворення Фур'є (ПФ):

$$F(\omega_x,\omega_y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(x,y)e^{-j(\omega_x x+\omega_y y)}dxdy\ , \tag{1.3}$$

де $\omega_x=2\pi\upsilon_x$ і $\omega_y=2\pi\upsilon_y$ – циклічні просторові частоти; υ_x і υ_y – звичайні просторові частоти, які мають фізичну розмірність [1/м]; $F(\omega_x,\omega_y)$ – комплексна двовимірна функція, яку представляють її реальною та уявною частинами, або її модулем та фазою. Реальна та уявна частини, а також модуль та фаза просторового спектра є двовимірними функціями аргументів ω_x і ω_y. Обернене двовимірне ПФ дозволяє отримати неперервне напівтонове зображення за заданим просторовим спектром. Обернене двовимірне ПФ є:

$$f(x,y)=\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(\omega_x,\omega_y)e^{j(\omega_x x+\omega_y y)}d\omega_x d\omega_y\ . \tag{1.4}$$

Інтегральні перетворення (1.3), (1.4) є лінійними. Це означає, що при їх застосуванні виконується принцип суперпозиції. Зокрема, ПФ від суми двох напівтонових зображень дорівнює сумі ПФ від кожного зображення. Перетворення (1.3) і (1.4) мають також набір спеціальних властивостей, які у більшості випадків аналогічні властивостям одновимірних ПФ [2]. При застосуванні перетворень (1.3), (1.4) варто враховувати таке. Якщо напівтонове зображення є обмеженим у просторі, то його просторовий спектр не є обмеженим

у просторово-частотній області. Це означає, що за умови строгої просторової фінітності функції f(x,y) будуть відсутні ненульові за площею області просторових частот, на яких функція F(x, y) тотожно дорівнює нулю. Аналогічно, якщо просторовий спектр напівтонового зображення є обмеженим у просторово-частотній області, то саме зображення є необмеженим у просторі. Також на основі поняття періодичного повторення двовимірної функції [2] можна встановити, що напівтонове зображення, яке періодично повторюється у просторі, має дискретний просторовий спектр. І навпаки, якщо просторовий спектр зображення є дискретним, то зображення буде періодично або майже періодично повторюватися у просторі.

Імовірнісний опис зображень

З інформаційної точки зору усі зображення можна поділити на два класи: детерміновані та випадкові. Детермінованими називають такі зображення, про які заздалегідь відомо усе. Це означає, що будь-яке значення функції, яка описує таке зображення, апріорно відоме або може бути передбачене з імовірністю, що дорівнює одиниці. Детерміновані зображення не мають корисної інформації і їх звичайно використовують як еталони. Випадковими називають зображення, значення яких заздалегідь невідомі і можуть бути тільки передбачені з імовірністю, що є меншою за одиницю. Усі реальні зображення варто вважати випадковими. Причинами цього можуть

бути як випадковий характер безпосередньо джерела зображення, так і випадковий характер процесу формування зображення, а також наявність шуму на зображенні. Будь-яке зображення, що має в собі корисну інформацію, повинне розглядатися як випадкове. Для опису детермінованих зображень використовують вирази, до яких входять різні композиції відомих (детермінованих) математичних функцій. Тому усі характеристики детермінованих зображень є детермінованими і можуть бути подані аналітично або чисельно.

Найзагальніший спосіб опису випадкового зображення – імовірнісний опис – заснований на заданні спільної щільності імовірності значень випадкової функції [2]. Нехай запис f(x,y) позначає випадкову неперервну функцію просторових координат x і y, яка породжує випадкове статичне напівтонове зображення. У загальному випадку випадковий процес f(x,y) цілком описується спільною щільністю імовірності $p(f_1,f_2,\ldots,f_N;x_1,y_1,x_2,y_2,\ldots,x_N,y_N)$ для N значень функції fi(xi,yi) у точках xi,yi. Але спільні щільності імовірності високого порядку для зображень зазвичай невідомі, і їх у загальному випадку важко моделювати [2]. Тому часто обмежуються більш простими моделями. Так, для щільності імовірності першого порядку p(f; x, y) іноді вдається підібрати вдалу модель з фізичних міркувань чи на основі обмірюваних гістограм. Наприклад, як моделі щільності імовірності яскравості застосовують [2]: закон розподілу Релея

$$p(f; x, y) = \frac{f(x, y)}{\alpha^2} \exp\{-\frac{[f(x, y)]^2}{2\alpha^2}\},$$

логарифмічно нормальний закон

$$p(f; x, y) = \frac{1}{\sqrt{2\pi} f(x, y)\sigma(x, y)} \exp\{-\frac{[\log(f(x, y)) - m(x, y)]^2}{2\sigma^2(x, y)}\}$$

та закон експоненціального розподілу імовірності:

$$p(f; x, y) = \alpha \exp\{-\alpha \mid f(x, y) \mid\}.$$

Це обумовлено тим, що яскравість може приймати тільки невід'ємні значення і тому щільність розподілу імовірності повинна бути однобічною. Для опису випадкового зображення можуть бути використані й умовні щільності імовірності. Так, умовна щільність розподілу імовірності значення функції f(x,y) у точці (x1,y1) при заданому значенні цієї функції в точці (x2,y2) визначається співвідношенням:

$$p(f_1; x_1, y_1 \mid f_2; x_2, y_2) = p(f_1, f_2; x_1, y_1, x_2, y_2) / p(f_2; x_2, y_2).$$

Аналогічно визначаються умовні щільності більш високого порядку [2]. Інший спосіб опису випадкового зображення полягає в обчисленні певних середніх (за ансамблем реалізацій) величин [2].

При цьому звичайно обмежуються обчисленнями першого і другого моментів. Так, перший початковий момент, або середнє значення функції f(x,y), є [2]:

$$m(x,y) = E[f(x,y)] = \int_{-\infty}^{\infty} f(x,y)\, p(f;x,y)\, df .$$

Другий початковий момент, або автокореляційна функція1, є [2]:

$$R(x_1,y_1;x_2,y_2) = E[f(x_1,y_1)\, f^*(x_2,y_2)] =$$

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x_1,y_1)\, f^*(x_2,y_2)\, p(f_1, f_2; x_1, y_1, x_2, y_2)\, df_1 df_2 .$$

Автоковаріаційна функція зображення обчислюється за формулою:

$$K(x_1,y_1;x_2,y_2) = E\{[f(x_1,y_1) - m(x_1,y_1)][f^*(x_2,y_2) - m^*(x_2,y_2)]$$

і є другим центральним моментом. Автокореляційна й автоковаріаційна функції зв'язані співвідношенням:

$$K(x_1,y_1;x_2,y_2) = R(x_1,y_1;x_2,y_2) - m(x_1,y_1)\, m^*(x_2,y_2) .$$

Середньоквадратичне значення випадкового процесу f(x,y) отримується при $x=x_1=x_2$, $y=y_1=y_2$ і $E[|f(x,y)|^2] = R(x,y;x,y)$, а дисперсія процесу $f(x,y)$ є: $\sigma^2(x,y) = K(x,y;x,y)$.

Випадковий процес, який народжує зображення, називають просторово-стаціонарним у строгому розумінні, якщо всі його моменти не залежать від зсуву початку системи координат у просторі. Випадковий процес називають просторово-стаціонарним у широкому сенсі, якщо зазначена вимога поширюється тільки на перші й другі моменти. Випадкове зображення, яке стаціонарне в строгому чи в широкому розумінні, має постійну середню яскравість, а його автокореляційна функція залежить тільки від різниці координат x_1-x_2, y_1-y_2, , але не залежить від самих координат, тобто $m(x, y)$=const; $R(x_1,x_2, y_1,y_2) = R(x_1$-$x_2, y_1$-$y_2)$. Тоді вираз для просторової автокореляційної функції можна записати так $R(\tau_x,\tau_y) = E[\, f(x+\tau_x, y+\tau_y)\, f^*(x, y)]$. Для спрощення описів автокореляційну функцію часто представляють у вигляді добутку автокореляційних функцій для кожної просторової змінної $R(\tau_x,\tau_y) = R_x(\tau_x) R_y(\tau_y)$. У зображеннях, які створені людиною, часто зустрічаються горизонтальні і вертикальні структури і, тому, таке представлення виявляється цілком прийнятним [2].

Часто опис випадкових зображень виконують за допомогою виразів, до яких входять композиції детермінованих і випадкових математичних функцій. Найпростішим прикладом є адитивна модель зображення у вигляді суми детермінованого зображення і випадкового просторового шуму. Іншим прикладом є адитивна

модель у вигляді суми випадкового зображення і випадкового просторового шуму, які є некорельованими випадковими процесами.

Контрольні питання

1. Які види зображень Ви знаєте і як їх можна формалізувати?

2. У чому відмінність моделей кольорового та напівтонового зображень?

3. За якою схемою виконується просторова обробка кольорових зображень?

4. Наведіть можливі 22 міжкомпонентної обробки варіанти багатокомпонентного зображення.

5. Чому для опису випадкових зображень застосовуються однобічні щільності розподілу імовірності?

Лекція 8. Інтелектуальні комп’ютерно-інтегровані системи автоматизації і робототехніки

Сучасна робототехніка переходить від систем із жорстким програмуванням до інтелектуальних робототехнічних комплексів, що здатні:

- самостійно сприймати середовище,
- будувати його модель,
- планувати дії,
- адаптуватися до змін,
- навчатись на даних,
- працювати автономно.

Це вимагає об’єднання:

1. Обробки інформації та сигналів,
2. Теорії автоматичного керування,
3. Комп’ютерного зору,
4. Машинного навчання,
5. Розподілених обчислень,
6. Сенсорного злиття,
7. Робототехнічних алгоритмів.

8.2. Структура комп’ютерно-інтегрованої системи робототехніки

8.2.1. Типова архітектура

1. Сенсорний шар

   -камери, LiDAR, IMU, енкодери, радар, тактильні датчики.

2. Шар попередньої обробки

   -фільтрація сигналів, корекція шумів, нормалізація.

3. Аналітичний шар

   -ML-моделі, розпізнавання об’єктів, класифікація станів.

4. Алгоритмічний шар

   -локалізація, SLAM, планування руху.

5. Шар контролю

   -регулятори, оптимізаційні закони керування.

6. Виконавчий шар

   -приводи, сервомотори, маніпулятори, платформи.

8.2.2. Концепція цифрового робота

Сучасні роботи формують цифрову структуру:

Data→Models→Decisions→Actions\text{Data} \rightarrow \text{Models} \rightarrow \text{Decisions} \rightarrow \text{Actions}Data→Models→Decisions→Actions

Дані → Моделі → Рішення → Дії.

8.3. Роль теорії автоматичного керування в інтелектуальних роботах

Навіть найскладніші AI-системи працюють поверх класичних контурів керування.

8.3.1. Рівні керування

- Нижній рівень — ПІД, адаптивні фільтри, моделі приводів.
- Середній рівень — алгоритми стабілізації: ЛКК, MPC.
- Верхній рівень — планування, зворотна кінематика, поведінкова логіка.

8.3.2. Інтелектуальні регулятори

На базі ML:

- Нейроконтролери
- Fuzzy-контролери
- Нейро-фаззі системи
- Learning-Based Control
- Reinforcement Learning Control (керування з підкріпленням)

8.4. Інтеграція ML у робототехнічні системи

8.4.1. Використання ML в роботах

- Класифікація об’єктів
- Виявлення аномалій у сигналах
- Прогноз часу відмови ("predictive maintenance")
- Управління роботом з підкріпленням
- Роботизоване сприйняття (Perception)
- автономне прийняття рішень

8.4.2. Типові моделі

- CNN (VGG, ResNet, YOLO, EfficientNet)
- RNN / LSTM для часових сигналів
- Transformer (ViT, DETR)
- Graph Neural Networks для SLAM
- Autoencoder для стискання вимірювань

8.4.3. Живучість і відмовостійкість ML-систем

Проблеми:

- шумні сигнали
- змінні умови (освітлення, температура, вібрації)
- чутливість до калібровки
- обчислювальні обмеження

Методи підвищення надійності:

- Ensemble models
- Data augmentation
- Sensor fusion
- Робастні оцінювачі
- Online learning

8.5. Системи навігації та локалізації (SLAM)

SLAM — одночасна локалізація й побудова карти.

8.5.1. Типи SLAM

- LiDAR-SLAM (Hector SLAM, GMapping)
- Visual SLAM (ORB-SLAM, LSD-SLAM)
- RGB-D SLAM
- Stereo SLAM
- Tightly-coupled VIO (VINS-Mono, OKVIS)
- Graph-based SLAM

8.5.2. Етапи SLAM

1. Вилучення ознак (feature extraction)
2. Трекінг ознак

3. Оцінка руху

4. Побудова локальної карти

5. Глобична оптимізація графа (loop closure)

8.5.3. Приклади використання

- дрони (VIO + GPS)
- мобільні платформи
- автономні машини
- AGV на складах
- роботи-прибиральники
- інспекційні роботи

8.6. Планування руху

Робот повинен генерувати траєкторію з урахуванням:

- фізичних обмежень
- кількості степенів свободи
- динаміки
- перешкод у середовищі

8.6.1. Планувальники

- A*

- D*
- RRT / RRT*
- PRM
- MPC-based planners
- Artificial Potential Fields

8.6.2. Планування для маніпуляторів

- обчислення зворотної кінематики
- врахування сингулярностей
- уникнення зіткнень
- оптимізація енерговитрат
- забезпечення плавності траєкторії

8.6.3. Планування у 3D-середовищі

Застосовується у:

- дронах
- автономних автомобілях
- мобільних роботах з маніпуляторами

8.7. Інтелектуальні системи контролю якості

Комп’ютерний зір застосовується:

- у виробництві
- в медицині
- у безпеці
- в агротехнологіях

8.7.1. Розпізнавання дефектів

Моделі:

- CNN для класифікації
- YOLO/SAM для сегментації
- Autoencoder для аномалій
- GAN для генерації дефектів

8.7.2. 3D-зір

Методи:

- стереозір
- LiDAR-ф’южн
- глибінні камери (ToF)
- структуроване світло

8.8. Роботизоване відчуття (Multimodal Perception)

Сучасні роботи комбінують:

- візуальну інформацію
- інерціальну
- аудіо
- радарні дані
- тактильні масиви

8.8.1. Тактильний зір

Приклади:

- GelSight
- DIGIT
- BioTac

8.8.2. Мультисенсорний контроль захвату

Алгоритми:

- аналітичні моделі сили
- ML-підхід: CNN+LSTM
- reinforcement learning gripping
- force feedback control

8.9. Цифрові двійники та симуляція

Симулятори:

- Gazebo
- Webots
- Isaac Sim
- CoppeliaSim
- Unity Robotics
- AirSim

Вони дозволяють:

- тестувати ML-моделі
- будувати кібер-фізичні системи
- моделювати поведінку роботів у складних умовах
- відпрацьовувати керування

8.10. Архітектура промислової комп’ютерно-інтегрованої системи

Рівні:

1. PLC-рівень — нижня автоматика
2. SCADA — операторські панелі
3. MES — диспетчеризація
4. ERP — бізнес-рівень
5. AI-рівень — аналітика, прогнозування, оптимізація

8.11. Безпека робототехнічних систем

- Надійність сенсорів
- Виявлення помилок управління
- Safe AI
- Захист каналів передачі
- Виявлення атак на ML

Лекція 9. Основи захисту інформації в комп’ютерно-інтегрованих та робототехнічних системах

9.1. Основні поняття

Наукові дослідження проблем захисту інформації, інформаційної безпеки були, є і будуть завжди актуальними. В процесі досліджень проблем захисту інформації в комп’ютерно-інтегрованих та робототехнічних системах виділяють такі властивості інформації, як: адресність, актуальність, можливість кодування, висока швидкість збору, обробки та передачі, достатність, достовірність, багаторазовість використання, правова коректність, повнота, своєчасність.

Під інформаційною безпекою або безпекою інформації (Information security) розуміють захищеність інформації та підтримуючої інфраструктури від випадкових чи навмисних впливів природного чи штучного характеру, здатних завдати шкоди власникам та користувачам інформації та структурі, що її підтримує.

При розгляді проблем, пов'язаних із забезпеченням безпеки, захисту інформації в комп’ютерно-інтегрованих та робототехнічних системах, використовують поняття несанкціонований доступ – неправомірне звернення до інформаційних ресурсів з метою їх

використання (читання, модифікації), а також псування чи знищення.

У свою чергу поняття санкціонований доступ означатиме доступ до об'єктів, програм та даних користувачів, які мають право виконувати певні дії (читання, копіювання та ін.), а також повноваження та права користувачів на використання ресурсів та послуг, визначених адміністратором обчислювальної системи.

Захищеною вважають інформацію, яка не зазнала незаконних змін у процесі передачі, обробки та зберігання, яка не змінила свої основні властивості (конфіденційність, цілісність, доступність). Порушення кожної з трьох складових призводить до порушення інформаційної безпеки в цілому.

Отже, під терміном захист інформації мається на увазі сукупність методів, засобів та заходів, спрямованих на виключення спотворень, знищення та несанкціонованого використання накопичуваних, оброблюваних та збережених даних, тобто відсутності змін основних властивостей інформації.

9.2. Загальні поняття захисту інформації в автоматизованих системах

Закон України “Про захист інформації в автоматизованих системах” поширюється на будь-яку інформацію (кіберінформацію),

що обробляється в автоматизованих системах (АС).

Закон визначає: «захист інформації – сукупність організаційно-технічних заходів і правових норм для запобігання заподіянню шкоди інтересам власника інформації чи АС та осіб, які користуються інформацією».

Під АС розуміється «система, що здійснює автоматизовану обробку даних і до складу якої входять технічні засоби їх обробки (засоби обчислювальної техніки і зв'язку), а також методи і процедури, програмне забезпечення».

Отже, об'єктами захисту є:

• інформація, що обробляється в АС,

• права власників цієї інформації та власників АС,

• права користувача.

Інформація, яка є власністю держави, або інформація, захист якої гарантується державою, повинна оброблятись в АС, що має відповідний сертифікат (атестат) захищеності.

Комп’ютерно-інтегровані та робототехнічні системи це автоматизовані системи обробки інформації, які складаються з наступних взаємопов'язаних компонентів:

- технічних засобів обробки та передачі даних (засобів обчислювальної техніки та зв'язку);

- методів та алгоритмів обробки у вигляді відповідного програмного забезпечення;
- інформації (масивів, наборів, баз даних) на різних носіях;
- обслуговуючого персоналу та користувачів системи, об'єднаних за організаційно-структурною, тематичною, технологічною або іншими ознаками для виконання автоматизованої обробки інформації (даних) з метою задоволення інформаційних потреб суб'єктів інформаційних відносин.

Обробкою інформації в комп’ютерно-інтегрованих та робототехнічних системах називається будь-яка сукупність операцій (прийом, накопичення, зберігання, перетворення, відображення, передача тощо), що здійснюються над інформацією (відомими даними) з використанням засобів комп’ютерно-інтегрованих та робототехнічних систем.

Суб'єктами називатимемо:

- держава (в цілому або окремі її органи та організації);
- громадські чи комерційні організації (об'єднання) та підприємства (юридичних осіб);
- окремих громадян (фізичних осіб).

У процесі своєї діяльності суб'єкти можуть бути один з одним у різного роду відносинах, у тому числі, що стосуються питань отримання, зберігання, обробки, поширення та використання певної інформації. Такі відносини між суб'єктами називатимемо

інформаційними відносинами, а самих суб'єктів, що беруть участь в них, - суб'єктами інформаційних відносин.

Різні суб'єкти стосовно певної інформації можуть (і навіть одночасно) виступати як (у ролі):

- джерел (постачальників) інформації;
- споживачів (користувачів) інформації;
- власників, розпорядників інформації;
- фізичних та юридичних осіб, про яких збирається інформація;
- власників систем обробки інформації;
- учасників процесів обробки та передачі інформації тощо.

Захист інформації полягає не лише в захисті засобів обробки інформації, а в організації засобів захисту для підтримки певних властивостей інформації.

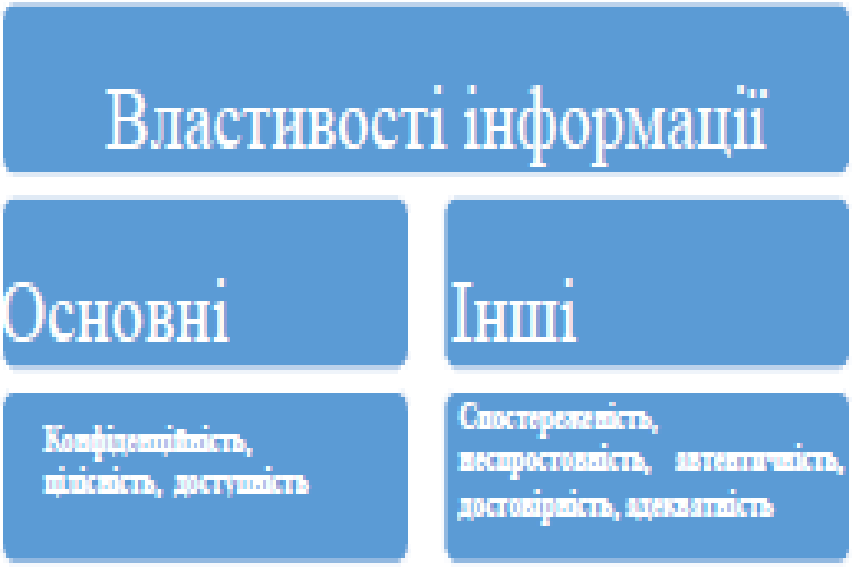


Рис. 9.1. Властивості інформації з точки зору її захисту

Виділяють наступні види властивостей інформації. Основними (фундаментальними) властивостями є конфіденційність, цілісність та доступність, адже захист інформації в більшості випадків пов'язаний з комплексним рішенням трьох завдань:

1) забезпеченням **конфіденційності** інформації.

2) забезпеченням **цілісності** інформації;

3) забезпеченням **доступності** інформації.

Для успішного здійснення своєї діяльності з управління об'єктами певної предметної галузі суб'єкти інформаційних відносин можуть бути зацікавлені у забезпеченні:

- своєчасного доступу (за прийнятний для них час) до необхідної їм інформації та певних автоматизованих служб;
- конфіденційності (збереження в таємниці) певної частини інформації;
- достовірності (повноти, точності, адекватності, цілісності) інформації;
- захисту від нав'язування ним хибної (недостовірної, спотвореної) інформації (тобто від дезінформації);
- захисту частини інформації від незаконного її тиражування (захисту авторських прав, прав власника інформації тощо);
- розмежування відповідальності за порушення законних прав (інтересів) інших суб'єктів інформаційних відносин та встановлених правил поводження з інформацією;
- можливості здійснення безперервного контролю та управління процесами обробки та передачі інформації тощо.

Будучи зацікавленим у забезпеченні хоча б однієї з вищезгаданих вимог, суб'єкт інформаційних відносин є вразливим, тобто потенційно схильним до заподіяння йому шкоди (прямої чи непрямої, матеріальної чи моральної) через вплив на критичну для нього інформацію, її носії та процеси обробки, або за допомогою неправомірного використання такої інформації. Тому всі суб'єкти інформаційних відносин тією чи іншою мірою (залежно від розмірів

шкоди, яку їм може бути завдано) зацікавлені у забезпеченні своєї інформаційної безпеки.

Оскільки збитки суб'єктам інформаційних відносин можуть бути завдані опосередковано, через певну інформацію та її носії, то закономірно виникає зацікавленість суб'єктів у забезпеченні безпеки цієї інформації, її носіїв та систем обробки.

Звідси випливає, що як об'єкти, що підлягають захисту для забезпечення безпеки суб'єктів інформаційних відносин, повинні розглядатися: інформація, будь-які її носії (окремі компоненти комп’ютерно-інтегрованих та робототехнічних систем та АС в цілому) та процеси обробки (передачі).

Проте завжди слід пам'ятати, що вразливими в кінцевому рахунку є саме зацікавлені в забезпеченні певних властивостей інформації та систем її обробки суб'єкти. Тому, говорячи про забезпечення безпеки в комп’ютерно-інтегрованих та робототехнічних системах або інформації, що циркулює в них, завжди слід розуміти під цим непряме забезпечення безпеки відповідних суб'єктів, що беруть участь у процесах автоматизованої інформаційної взаємодії.

Оскільки суб'єктам інформаційних відносин шкода може бути завдана також за допомогою впливу на процеси та засоби обробки критичної для них інформації, стає очевидною необхідність

забезпечення захисту системи обробки та передачі даної інформації від несанкціонованого втручання в процес її функціонування, а також від спроб розкрадання, незаконної модифікації та/або руйнування будь-яких компонентів даної системи.

9.3. Мета захисту комп’ютерно-інтегрованих та робототехнічних систем та інформації, що циркулює в них, загрози безпеці в автоматизованих системах

9.3.1. Особливості автоматизованих систем та мета захисту комп’ютерно-інтегрованих та робототехнічних систем

При розгляді проблем забезпечення інформаційної безпеки, захисту комп’ютерно-інтегрованих та робототехнічних систем слід завжди виходити з того, що захист інформації та обчислювальної системи її обробки не є самоціллю.

Кінцевою метою створення системи безпеки в комп’ютерно-інтегрованих та робототехнічних системах є захист усіх категорій суб'єктів, які прямо чи опосередковано беруть участь у процесах інформаційної взаємодії, від завдання їм відчутної матеріальної, моральної чи іншої шкоди внаслідок випадкових чи навмисних небажаних впливів на інформацію та системи її обробки та передачі.

Як об'єкти, що захищаються, повинні розглядатися інформація, всі її носії (окремі компоненти і автоматизована система обробки інформації в цілому) і процеси обробки.

Основним завданням системи захисту є забезпечення необхідного рівня доступності, цілісності та конфіденційності компонентів комп’ютерно-інтегрованих та робототехнічних систем.

Забезпечення інформаційної безпеки – це безперервний процес, основний зміст якого становить практична реалізація організаційно-технічних заходів. До таких заходів відносять управління людьми, ризиками, ресурсами, засобами захисту тощо. Люди – обслуговуючий персонал та кінцеві користувачі комп’ютерно-інтегрованих та робототехнічних систем, – є невід'ємною частиною автоматизованої (тобто «людино-машинної») системи. Від того, яким чином вони реалізують свої функції в системі, суттєво залежить не лише її функціональність (ефективність розв'язання задач), а й її безпека.

Метою захисту циркулюючої в комп’ютерно-інтегрованих та робототехнічних системах інформації (кіберінформації) є запобігання її розголошенню (витіканню), спотворенню (модифікації), втраті, блокуванню (зниженню ступеня доступності) або незаконному тиражуванню.

Забезпечення безпеки обчислювальної системи передбачає

створення перешкод для будь-якого несанкціонованого втручання у процес її функціонування, а також для спроб розкрадання, модифікації, виведення з ладу або руйнування її компонентів, тобто захист усіх компонентів системи: обладнання, програмного забезпечення, даних (кіберінформації) та її персоналу.

У цьому сенсі захист інформації від несанкціонованого доступу (НСД) є лише частиною загальної проблеми забезпечення безпеки комп’ютерно-інтегрованих та робототехнічних систем та захисту законних інтересів суб'єктів інформаційних відносин.

Одним із найважливіших аспектів проблеми забезпечення безпеки комп’ютерно-інтегрованих та робототехнічних систем є визначення, аналіз та класифікація можливих загроз безпеці в комп’ютерно-інтегрованих та робототехнічних системах. Перелік значних загроз, оцінки ймовірностей їх реалізації, а також модель порушника є основою для проведення аналізу ризиків та формулювання вимог до системи захисту в комп’ютерно-інтегрованих та робототехнічних системах.

Більшість сучасних комп’ютерно-інтегрованих та робототехнічних систем в загальному випадку є територіально розподіленими системами, що інтенсивно взаємодіють між собою за даними (кіберінформацією). У розподілених комп’ютерно-інтегрованих та робототехнічних системах можливі всі, так звані

традиційні, для локально розташованих обчислювальних систем способи несанкціонованого втручання у їхню роботу та доступу до інформації. Крім того, для них характерні і нові специфічні канали проникнення в систему та несанкціонованого доступу до кіберінформації, наявність яких пояснюється цілою низкою їх особливостей.

Перерахуємо основні особливості розподілених комп'ютерно-інтегрованих та робототехнічних систем**:**

- територіальна розосередженість складових систем та наявність інтенсивного обміну кіберінформацією між ними;
- широкий спектр використовуваних способів подання, зберігання та протоколів передачі кіберінформації;
- інтеграція даних різного призначення, що належать різним суб'єктам, у межах єдиних баз даних та, навпаки, розміщення необхідних деяким суб'єктам даних у різних віддалених вузлах мережі;
- абстрагування власників даних від фізичних структур та місця розміщення даних;
- використання режимів розподіленої обробки даних;
- участь у процесі автоматизованої обробки інформації великої кількості користувачів та персоналу різних категорій;
- безпосередній та одночасний доступ до ресурсів (у тому числі й інформаційних) великої кількості користувачів (суб'єктів)

різних категорій;

- високий ступінь різнорідності засобів обчислювальної техніки та телекомунікацій, що використовуються, а також їх програмного забезпечення;
- відсутність спеціальних засобів захисту у більшості типів технічних засобів, що широко використовуються в комп’ютерно-інтегрованих та робототехнічних системах.

Розглянемо основні складові комп’ютерно-інтегрованих та робототехнічних систем. Вони складаються з таких структурно-функціональних елементів:

- робочих станцій окремих комп’ютерно-інтегрованих систем чи терміналів мережі, у яких реалізуються автоматизовані робочі місця користувачів;
- серверів не виділених (або виділених, тобто не поєднаних із робочими станціями) високопродуктивних комп’ютерів, призначених для реалізації функцій зберігання, друку даних, обслуговування робочих станцій мережі тощо;
- мережевих пристроїв (маршрутизаторів, комутаторів, шлюзів, центрів комутації пакетів, комунікаційних комп’ютерів) - елементів, що забезпечують з'єднання кількох мереж передачі даних, або кількох сегментів однієї і тієї ж мережі;
- телекомунікаційних каналів зв'язку (кабельних, радіоканалів, телефонних, мультимедійних тощо).

Робочі станції є найбільш доступними компонентами мереж і саме з них можуть бути здійснені найбільш численні спроби вчинення несанкціонованих дій.

Особливого захисту потребують такі привабливі для зловмисників елементи мереж, як сервери (host-машини) та мережні пристрої. Перші - як концентратори великих обсягів інформації, другі - як елементи, в яких здійснюється перетворення (можливо через відкриту, незашифровану форму подання) даних за погодженням протоколів обміну в різних ділянках мережі.

Сприятливою для підвищення безпеки серверів і мостів обставиною є, як правило, наявність можливостей їх надійного захисту фізичними засобами та організаційними заходами через їх виділення, що дозволяє скоротити до мінімуму кількість осіб із персоналу, які мають безпосередній доступ до них. У той же час, все більш поширеними стають масовані атаки на сервери та мости (а також і на робочі станції) з використанням засобів віддаленого доступу. Тут зловмисники насамперед можуть шукати можливості вплинути на роботу різних підсистем робочих станцій, серверів та мостів, використовуючи недоліки протоколів обміну та засобів розмежування віддаленого доступу до ресурсів та системних таблиць. Використовуватись можуть усі можливості та засоби, від стандартних (без модифікації компонентів) до підключення спеціальних апаратних засобів (канали, як правило, слабо захищені

від підключення) та застосування висококласних програм для подолання системи захисту.

Канали та засоби зв'язку також потребують захисту. Через велику просторову довжину ліній зв'язку (через неконтрольовану або слабко контрольовану територію) практично завжди існує можливість підключення до них, або втручання в процес передачі даних.

Важливим чинником, що впливає на захист інформації, є передача через Інтернет з одночасним забезпеченням безпеки цих комунікацій. Вирішення проблем інформаційної безпеки, пов'язаних з широким поширенням Internet, Intranet та Extranet це одне з найактуальніших завдань, що стоїть перед розробниками та постачальниками інформаційних технологій.

Завдання забезпечення інформаційної безпеки вирішується побудовою системи інформаційної безпеки (СІБ). СІБ, що створюється, повинна враховувати появу нових технологій і сервісів, а також задовольняти загальним вимогам, таких які стосуються застосування відкритих стандартів, використання інтегрованих режимів, забезпечення масштабування.

Перехід на відкриті стандарти становить одну з основних тенденцій розвитку засобів інформаційної безпеки. Такі стандарти як IPSec та PKI забезпечують захищеність зовнішніх комунікацій

підприємств та сумісність із відповідними продуктами підприємств-партнерів та віддалених клієнтів. Цифрові сертифікати X.509 також є на сьогодні стандартною основою для автентифікації користувачів та пристроїв. Перспективні засоби захисту, безумовно, мають підтримувати ці стандарти сьогодні.

Під інтегрованими рішеннями розуміється як інтеграція засобів захисту з іншими елементами мережі (ОС, маршрутизаторами, службами каталогів тощо), і інтеграція різних технологій безпеки між собою забезпечення комплексного захисту інформаційних ресурсів підприємств, наприклад інтеграція міжмережевого екрану з VPN-шлюзом і транслятором IP-адрес.

У міру зростання та розвитку системи інформаційної безпеки вона повинна мати можливість легко масштабуватись без втрати цілісності та керованості. Масштабованість засобів захисту дозволяє підбирати оптимальне за вартістю та надійністю рішення з можливістю поступового нарощування системи захисту. Масштабування має забезпечувати ефективну роботу підприємств за наявності у нього численних філій, підприємств-партнерів, сотень віддалених співробітників та мільйонів клієнтів.

Для того, щоб забезпечити надійний захист ресурсів в комп’ютерно-інтегрованих та робототехнічних системах, у системи інформаційної безпеки мають бути реалізовані найпрогресивніші та

найперспективніші технології інформаційного захисту. До них відносяться:

- криптографічний захист даних для забезпечення конфіденційності, цілісності та достовірності інформації;
- технології аутентифікації для перевірки справжності користувачів та об'єктів;
- технології міжмережевих екранів для захисту локальної мережі від зовнішніх загроз при підключенні до загальнодоступних мереж зв'язку;
- технології віртуальних захищених каналів та мереж VPN для захисту інформації, що передається по відкритих каналах зв'язку;
- гарантована ідентифікація користувачів шляхом застосування токенів (смарт-карт, touch-memory, ключів для USB-портів тощо) та інших засобів автентифікації;
- підтримка інфраструктури управління відкритими ключами VPN;
- управління доступом на рівні користувачів та захист від несанкціонованого доступу до інформації;
- технології виявлення вторгнень (Intrusion Detection) для активного дослідження захищеності інформаційних ресурсів;
- технології захисту від вірусів з використанням спеціалізованих комплексів антивірусної профілактики та захисту;

– централізоване управління СІБ на базі єдиної безпекової політики підприємства;
– комплексний підхід до забезпечення інформаційної безпеки, що забезпечує раціональне поєднання технологій та засобів інформаційного захисту.

9.3.2. Загрози безпеці, джерела загроз безпеці в комп'ютерно-інтегрованих та робототехнічних системах

Розгляд можливих загроз інформаційної безпеки проводиться з метою визначення повного набору вимог до системи захисту.

Перелік загроз, оцінки ймовірностей їх реалізації, а також модель порушника є основою для аналізу ризику реалізації загроз та формулювання вимог до системи захисту в комп'ютерно-інтегрованих та робототехнічних системах. Крім виявлення можливих загроз, доцільним є також проведення аналізу цих загроз на основі їхньої класифікації за низкою ознак. Кожна з ознак класифікації відображає одну із узагальнених вимог до системи захисту.

Знання можливих загроз, а також вразливих місць захисту, які ці загрози зазвичай експлуатують, необхідне для того, щоб вибирати ефективні засоби забезпечення безпеки.

Зазначалось раніше про те, що безпечна система повинна мати

властивості конфіденційності, доступності та цілісності. Будь-яка потенційна дія або подія, яка спрямована на порушення конфіденційності, цілісності та доступності інформації, називається **загрозою безпеці інформації**. Реалізована загроза безпеці інформації називається **атакою**.

Конфіденційна (confidentiality) система забезпечує впевненість у тому, що секретні дані будуть доступні лише тим користувачам, яким цей доступ дозволено (такі користувачі називаються авторизованими). Під доступністю (availability) розуміють гарантію того, що авторизованим користувачам завжди буде доступна інформація, яка їм потрібна. І нарешті, цілісність (integrity) системи передбачає, що неавторизовані користувачі що неспроможні якимось чином модифікувати дані.

Захист інформації орієнтований на боротьбу з так званими навмисними загрозами (рис. 14.2), тобто з тими, які, на відміну від ненавмисних випадкових загроз (помилок користувача, збоїв обладнання, неправильних дій користувачів або адміністрації та інші), мають на меті завдати шкоди користувачам систем.

Рис. Загрози безпеці інформації

Рис. 9.2. Загрози безпеці інформації

Усі загрози можуть бути реалізовані лише за наявності якихось слабких місць вразливостей, властивих об'єкту інформатизації. Вразливість - якась слабкість, яку можна використовувати для

порушення захисту комп’ютерно-інтегрованих та робототехнічних систем та інформації, що циркулює в них.

Особлива увага при розгляді інформаційної безпеки має приділятися джерелам загроз, в якості яких можуть виступати як суб'єкти (особистість), так і об'єктивні прояви. Причому самі джерела загроз можуть бути як всередині об'єкта інформатизації – внутрішні, так і поза його – зовнішні.

Як джерела загроз можуть бути: дії суб'єкта (антропогенні джерела загроз); технічні засоби (техногенні джерела небезпеки). До антропогенних джерел загроз належать суб'єкти, дії яких можуть бути кваліфіковані як навмисні чи випадкові **злочини.**

До техногенних джерел загроз належать джерела, що визначаються технократичною діяльністю людини та розвитком цивілізації. До стихійних джерел загроз належать стихійні лиха чи інші обставини, які неможливо чи можливо передбачити, але неможливо запобігти за сучасного рівня людського знання та можливостей.

Найбільших збитків інформації в комп’ютерно-інтегрованих та робототехнічних системах завдають неправомірні дії співробітників та комп'ютерні віруси. Вирішення цих проблем належить до компетенції адміністрації та служби безпеки організації. При цьому рекомендується шифрувати навіть листування, яке відбувається в

організації.

Віруси представляють широко поширене явище, що відбивається більшості користувачів комп'ютерів в комп’ютерно-інтегрованих та робототехнічних системах . Віруси з'явилися в результаті створення програм,. Віруси - це клас програм, які незаконно проникають у комп'ютери і завдають шкоди їх програмному забезпеченню, інформаційним файлам і навіть технічним пристроям. З розвитком інформаційних технологій віруси стали становити реальну загрозу для користувачів мережевих та локальних комп'ютерних систем.

Програма-вірус зазвичай складається з унікальної послідовності команд - сигнатур (знаків) і типових алгоритмів дій, що дозволяє створювати програми-антивіруси, які виявляють їх. Деякі віруси не мають унікальних сигнатур і типових алгоритмів дій й можуть видозмінювати себе (так звані, поліморфні віруси).

Типовими причинами порушення безпеки на об'єкті є: помилки людей або неточні дії; несправність та (або) відмова устаткування, що використовується; непередбачувані та неприпустимі зовнішні прояви; несправність та (або) відсутність необхідних засобів захисту; випадкові та навмисні впливи на інформацію, елементи обладнання, що захищаються, людини і навколишнє середовище.

Як було зазначено раніше, **загрози безпеці** можуть бути поділені

на два види: внутрішні та зовнішні. **Внутрішні загрози** безпеці об'єкта захисту: некваліфікована політика щодо організації інформаційних технологій та управління безпекою; відсутність належної кваліфікації персоналу щодо управління об'єктом захисту; навмисні та ненавмисні дії персоналу з порушення безпеки тощо.

Зовнішні загрози безпеці об'єкта захисту: негативні впливи конкурентів та державних структур; навмисні та ненавмисні дії зацікавлених структур та осіб (збір інформації, шантаж, спотворення іміджу, загрози фізичного впливу та інше); витік конфіденційної інформації з носіїв інформації та каналів зв'язку; несанкціоноване проникнення на об'єкт захисту; несанкціонований доступ до носіїв інформації та каналів зв'язку з метою розкрадання, спотворення, знищення, блокування інформації; стихійні лиха та інші форс-мажорні обставини; навмисні та ненавмисні дії системних інтеграторів та постачальників послуг із забезпечення безпеки, постачальників технічних та програмних продуктів тощо.

При комплексному підході до аналізу загроз інформаційної безпеки об'єкта інформатизації необхідно провести: опис об'єкта; класифікацію джерел загроз; класифікацію вразливостей; класифікацію методів реалізації загроз; ранжування актуальних атак; класифікацію методів упередження загроз.

Структурований опис об'єкта інформатизації, представлений у

вигляді типових структурних компонентів інформаційної системи та зв'язків між ними, що характеризують напрями циркуляції та параметри потоків інформації разом із текстовими поясненнями, дозволяє виявити точки можливого застосування загроз або розкрити існуючі вразливості.

Аналіз та оцінка можливостей реалізації загроз мають бути засновані на побудові моделі загроз, класифікації, аналізі та оцінки джерел загроз, вразливостей та методів реалізації. Моделювання процесів порушення інформаційної безпеки може здійснюватися на основі розгляду логічного ланцюжка: загроза – джерело загрози – метод реалізації – вразливість – наслідки.

Загрози класифікуються за тим наскільки вони можуть заподіяти шкоди у разі порушення цілей інформаційної безпеки.

Класифікація атак є сукупністю можливих варіантів дій джерела загроз певними методами реалізації з використанням вразливостей, які призводять до реалізації цілей атаки. Мета атаки може не збігатися з метою реалізації загроз і бути спрямована на отримання проміжного результату, необхідного для досягнення подальшої реалізації загрози. У разі розбіжності цілей атаки з метою реалізації загрози сама атака розглядається як етап підготовки до здійснення дій, спрямованих на загрози.

На основі проведеної класифікації, ранжування, аналізу та

визначення актуальних загроз, джерел загроз та вразливостей визначаються варіанти можливих атак, які дозволяють оцінити стан інформаційної безпеки.

Заходи, що вживаються для забезпечення безпеки в комп'ютерно-інтегрованих та робототехнічних системах, повинні бути комплексними: від забезпечення фізичної недоступності складових систем до ретельної регламентації роботи користувачів.

9.3.3. Основні ненавмисні та навмисні штучні загрози

Основні **ненавмисні** штучні загрози в комп'ютерно-інтегрованих та робототехнічних системах (дії, скоєні людьми випадково, через незнання, неуважність або недбалість, з цікавості, але без злого наміру):

- ✓ ненавмисні дії, що призводять до часткової або повної відмови системи або руйнування апаратних, програмних, інформаційних ресурсів системи (ненавмисне псування обладнання, видалення, спотворення файлів з важливою інформацією або програм, у тому числі системних тощо);
- ✓ неправомірне відключення обладнання або зміна режимів роботи пристроїв та програм;
- ✓ ненавмисне псування носіїв інформації;

- ✓ запуск технологічних програм, здатних при некомпетентному використанні викликати втрату працездатності системи (зависання чи зациклювання) або які здійснюють незворотні зміни у системі (форматування чи реструктуризацію носіїв інформації, видалення даних тощо);
- ✓ нелегальне впровадження та використання неврахованих програм (ігрових, навчальних, технологічних та ін., які не є необхідними для виконання порушником своїх службових обов'язків) з подальшим необґрунтованим витрачанням ресурсів (завантаження процесора, захоплення оперативної пам'яті та пам'яті на зовнішніх носіях);
- ✓ зараження комп'ютера вірусами;
- ✓ необережні дії, що призводять до розголошення конфіденційної інформації або роблять її загальнодоступною;
- ✓ розголошення, передача або втрата атрибутів розмежування доступу (паролів, ключів шифрування, ідентифікаційних карток, перепусток тощо);
- ✓ проектування архітектури системи, технології обробки даних, розробка прикладних програм, з можливостями, що становлять небезпеку для працездатності системи та безпеки інформації;
- ✓ ігнорування організаційних обмежень (встановлених правил) під час роботи у системі;

- ✓ вхід у систему в обхід засобів захисту (завантаження сторонньої операційної системи зі змінних магнітних носіїв тощо);
- ✓ некомпетентне використання, налаштування чи неправомірне відключення засобів захисту персоналом служби безпеки;
- ✓ пересилання даних за помилковою адресою абонента (пристрою);
- ✓ введення помилкових даних;
- ✓ ненавмисне ушкодження каналів зв'язку.

Основні можливі шляхи **навмисної** дезорганізації роботи, виведення системи з ладу, проникнення в систему та несанкціонованого доступу до інформації:

- ✓ фізичне руйнування системи (шляхом вибуху, підпалу тощо) або виведення з ладу всіх чи окремих найважливіших компонентів комп'ютерної системи (пристроїв, носіїв важливої інформації, осіб із числа персоналу тощо);
- ✓ відключення або виведення з ладу підсистем забезпечення функціонування обчислювальних систем (електроживлення, охолодження та вентиляції, ліній зв'язку тощо);
- ✓ дії щодо дезорганізації функціонування системи (зміна режимів роботи пристроїв або програм, страйк, саботаж персоналу, постановка потужних активних радіоперешкод на частотах роботи пристроїв системи тощо);

- ✓ впровадження агентів до персоналу системи;
- ✓ вербування персоналу або окремих користувачів, які мають певні повноваження;
- ✓ застосування підслуховуючих пристроїв, дистанційна фото- та відеозйомка тощо;
- ✓ перехоплення побічних електромагнітних, акустичних та інших випромінювань пристроїв та ліній зв'язку, а також наведення активних випромінювань на допоміжні технічні засоби, які безпосередньо не беруть участі в обробці інформації (телефонні кабелі, лінії електроживлення тощо);
- ✓ перехоплення даних, що передаються каналами зв'язку, та їх аналіз з метою з'ясування протоколів обміну, правил входження у зв'язок та авторизації користувача та подальших спроб їх імітації для проникнення в систему;
- ✓ розкрадання носіїв інформації (магнітних дисків, стрічок, мікросхем пам'яті);
- ✓ несанкціоноване копіювання носіїв інформації;
- ✓ розкрадання виробничих відходів;
- ✓ читання залишкової інформації з оперативної пам'яті та із зовнішніх запам'ятовуючих пристроїв;
- ✓ читання інформації з областей оперативної пам'яті, що використовуються операційною системою (у тому числі підсистемою захисту) або іншими користувачами, в асинхронному

режимі використовуючи недоліки мультизадачних операційних систем та систем програмування;

- ✓ розшифрування шифрів криптозахисту інформації;
- ✓ впровадження апаратних «закладок», програмних «закладок» та «вірусів»;
- ✓ незаконне підключення до ліній зв'язку з метою прямої заміни законного користувача шляхом його фізичного відключення після входу в систему та успішної аутентифікації з подальшим введенням дезінформації та нав'язуванням хибних повідомлень.

Слід зазначити, що найчастіше задля досягнення поставленої мети зловмисник використовує не один, а деяку сукупність із вище перерахованих шляхів.

9.3.4. Класифікація каналів проникнення в автоматизовану систему та витоку інформації із системи

Всі **канали проникнення** в систему та витоку інформації поділяють на прямі та непрямі. Під **непрямими** розуміють такі канали, використання яких вимагає проникнення в приміщення, де розташовані компоненти системи. Для використання прямих каналів таке проникнення потрібне.

Прямі канали можуть використовуватися без внесення змін до компонентів системи або змін компонентів. Канали проникнення

можна умовно розділити три групи. Класифікацію видів порушень працездатності систем та несанкціонованого доступу до інформації щодо об'єктів впливу та способів заподіяння шкоди безпеці наведено в таблиці 1.

За способом отримання інформації потенційні канали доступу можна поділити на:

- фізичний;
- електромагнітний (перехоплення випромінювань);
- інформаційний (програмно-математичний).

При контактному НСД (фізичному, програмно-математичному) можливі загрози інформації реалізуються шляхом доступу до елементів в комп’ютерно-інтегрованих та робототехнічних системах, до носіїв інформації, до інформації, що вводиться і виводиться (і результатів), до програмного забезпечення (у тому числі до операційних систем), а також шляхом підключення до ліній зв'язку.

При безконтактному доступі (наприклад, по електромагнітному каналу) можливі загрози інформації реалізуються перехопленням випромінювань апаратури в комп’ютерно-інтегрованих та робототехнічних системах, у тому числі наведених у струмопровідних комунікаціях і ланцюгах живлення, перехопленням інформації в лініях зв'язку, введенням в лінії зв'язку

хибної інформації, візуальним спостереженням.

Таблиця 9.1

Способи нанесення шкоди	Об'єкти впливів			
	Обладнання	Програми	Дані	Персонал
Розкриття (витік) інформації	Розкрадання носіїв інформації, підключення до лінії зв'язку, несанкціоноване використання ресурсів	Несанкціоноване копіювання перехоплення	Розкрадання, копіювання, перехоплення	Передача відомостей про захист, розголошення, недбалість
Втрата цілісності інформації	Підключення, модифікація, спецзакладки, зміна режимів роботи, несанкціоновані	Впровадження «троянських коней» та «жучків»	Спотворення, модифікація	Вербування персоналу

	використання ресурсів			
Порушення працездатності автоматизованої системи	Зміна режимів функціонування, виведення з ладу, розкрадання, руйнування	Спотворення, видалення, підміна	Спотворення, видалення, нав'язування хибних даних	Догляд, фізичне усунення
Незаконне тиражування інформації	Виготовлення аналогів без ліцензій	Використання незаконних копій	Публікація без відома авторів	

Злочини, зокрема й комп'ютерні, скоюються людьми. У цьому сенсі питання безпеки автоматизованих систем є суть питання людських стосунків та людської поведінки. Користувачі системи та її персонал, з одного боку, є складовою, необхідним елементом комп’ютерно-інтегрованих та робототехнічних систем. З іншого боку, вони є основною причиною і рушійною силою порушень і

злочинів.

Дослідження проблеми забезпечення безпеки комп'ютерних систем ведуться у напрямі розкриття природи явищ, що полягають у порушенні цілісності та конфіденційності інформації, дезорганізації роботи комп'ютерних систем. Серйозно вивчається статистика порушень, що викликають їх причини, особи порушників, суть застосовуваних порушниками прийомів та засобів, які використовуються при цьому недоліки систем та засобів їх захисту, обставини, за яких було виявлено порушення, та інші питання, які можуть бути використані при побудові моделей потенційних порушників.

Неформальна модель порушника відображає його практичні та теоретичні можливості, апріорні знання, час та місце дії тощо. Для досягнення своєї мети порушник повинен докласти деяких зусиль, витратити певні ресурси. Дослідивши причини порушень, можна або вплинути на самі ці причини (звісно, якщо це можливо), або точніше визначити вимоги до системи захисту від даного виду порушень або злочину.

Порушник - це особа, яка зробила спробу виконання заборонених операцій (дій) помилково, незнання або усвідомлено зі злим наміром (з корисливих інтересів) або без такого (заради гри чи задоволення, з метою самоствердження тощо) і використовує для

цього різні можливості, методи та засоби

Зловмисником називатимемо порушника, який навмисно йде на порушення з корисливих спонукань.

Під час розробки моделі порушника визначаються:

- ❖ припущення про категорії осіб, до яких може належати порушник;
- ❖ припущення про мотиви дій порушника (переслідуваних порушником цілей);
- ❖ припущення про кваліфікацію порушника та його технічну оснащеність (про методи та засоби, що використовуються для вчинення порушення);
- ❖ обмеження та припущення про характер можливих дій порушників.

По відношенню до комп'ютерно-інтегрованих та робототехнічних систем порушники можуть бути **внутрішніми** (з числа персоналу та користувачів системи) або **зовнішніми**. Внутрішнім порушником може бути особа з наступних категорій працівників:

- кінцеві користувачі (оператори);
- персонал, який обслуговує технічні засоби (інженери, техніки);
- співробітники відділів розробки та супроводу ПЗ

(прикладні та системні програмісти);

- співробітники служби безпеки комп’ютерно-інтегрованих та робототехнічних систем;
- керівники різних рівнів.

Сторонні особи, які можуть бути порушниками:

- технічний персонал, який обслуговує будівлі (прибиральники, електрики, сантехніки та інші співробітники, які мають доступ до будівель та приміщень, де розташовані компоненти комп’ютерно-інтегрованих та робототехнічних систем);
- клієнти (представники організацій, громадяни);
- відвідувачі (запрошені з будь-якого приводу);
- представники організацій, що взаємодіють з питань забезпечення життєдіяльності організації (енерго-, водо-, теплопостачання тощо);
- представники конкуруючих організацій (іноземних спецслужб) або особи, які діють за їх завданням;
- особи, які випадково або навмисне порушили пропускний режим (без мети порушити безпеку комп’ютерно-інтегрованих та робототехнічних систем);
- будь-які особи за межами контрольованої території.

Можна виділити кілька основних мотивів порушень:

- ✓ безвідповідальність;
- ✓ самоствердження;

- ✓ вандалізм;
- ✓ примус;
- ✓ помста;
- ✓ корисливий інтерес;
- ✓ ідейні міркування.

Порушення безпеки в комп’ютерно-інтегрованих та робототехнічних системах може бути пов'язане з примусом, помстою, ідейними міркуваннями чи корисливими інтересами користувача системи.

За **рівнем знань** про комп’ютерно-інтегровані та робототехнічні системи порушників можна класифікувати так:

– знає функціональні особливості комп’ютерно-інтегрованих та робототехнічних систем, основні закономірності формування у ній масивів даних та потоків запитів до них, вміє користуватися штатними засобами;

– має високий рівень знань та досвід роботи з технічними засобами системи та їх обслуговування;

– має високий рівень знань у галузі програмування та обчислювальної техніки, проектування та експлуатації автоматизованих інформаційних систем;

– знає структуру, функції та механізм дії засобів захисту, їх сильні та слабкі сторони.

За **рівнем можливостей** (використовуваних методів та засобів):

- застосовує лише агентурні методи отримання відомостей;
- застосовує, пасивні засоби (технічні засоби перехоплення без модифікації компонентів системи);
- використовує лише штатні засоби та недоліки систем захисту для її подолання (несанкціоновані дії з використанням дозволених засобів), а також компактні магнітні носії інформації, які можуть бути потай пронесені через пости охорони;
- застосовує, методи та засоби активної дії (модифікація та підключення додаткових технічних засобів, підключення до каналів передачі даних, впровадження програмних закладок та використання спеціальних інструментальних та технологічних програм).

За місцем дії:

- без доступу на контрольовану територію організації;
- з контрольованої території без доступу до будівель та споруд;
- усередині приміщень, але без доступу до технічних засобів комп'ютерно-інтегрованих та робототехнічних систем;
- із робочих місць кінцевих користувачів (операторів) в комп'ютерно-інтегрованих та робототехнічних системах;
- з доступом до зони даних (серверів баз даних, архівів тощо);
- з доступом до зони управління засобами забезпечення

безпеки в комп’ютерно-інтегрованих та робототехнічних системах.

Можуть враховуватися обмеження та припущення про характер дій можливих порушників, які пов’язані з добором кадрів.

9.4. Основні засади побудови системи захисту ресурсів в комп’ютерно-інтегрованих та робототехнічних системах

Побудова системи забезпечення безпеки інформації в комп’ютерно-інтегрованих та робототехнічних системах та їх функціонування мають здійснюватися відповідно до **основних принципів, які представлено нижче.**

Законність передбачає здійснення захисних заходів та розроблення системи безпеки інформації в комп’ютерно-інтегрованих та робототехнічних системах відповідно до чинного законодавства в галузі інформації, інформатизації та захисту інформації, інших нормативних актів з безпеки, затверджених органами державної влади.

Системний підхід до захисту інформації в комп’ютерно-інтегрованих та робототехнічних системах передбачає врахування всіх взаємопов'язаних, взаємодіючих та змінюваних у часі елементів, умов та факторів, суттєво значущих для розуміння та вирішення проблеми забезпечення інформаційної безпеки в комп’ютерно-

інтегрованих та робототехнічних системах.

При створенні системи захисту повинні враховуватися всі слабкі та найбільш вразливі місця системи обробки, зберігання та передавання інформації, а також характер, можливі об'єкти та напрямки атак на систем з боку порушників, шляхи проникнення у розподілені системи та НСД до інформації. Система захисту повинна будуватися з урахуванням не тільки всіх відомих каналів проникнення та НСД до інформації, а й з урахуванням можливості появи нових шляхів реалізації загроз безпеці.

Комплексне використання методів та засобів захисту в комп'ютерно-інтегрованих та робототехнічних системах передбачає узгоджене застосування різнорідних засобів при побудові цілісної системи захисту, що перекриває всі суттєві (значущі) канали реалізації загроз. Захист має будуватися ешелоновано. Зовнішній захист має забезпечуватись фізичними засобами, організаційними, технологічними та правовими заходами.

Одним з найбільш укріплених рубежів покликані бути засоби захисту, реалізовані на рівні операційних систем (ОС). Прикладний рівень захисту, що враховує особливості предметної області, становить внутрішній рубіж захисту.

Захист інформації - це безперервний цілеспрямований процес, що передбачає вжиття відповідних заходів на всіх етапах життєвого

циклу комп’ютерно-інтегрованих та робототехнічних систем, починаючи з ранніх стадій проектування, а не тільки на етапі її експлуатації.

Більшості фізичних та технічних засобів захисту для ефективного виконання своїх функцій потрібна **постійна** організаційна (адміністративна) підтримка. Перерви в роботі засобів захисту можуть бути використані зловмисниками для аналізу застосовуваних методів та засобів захисту, для впровадження спеціальних програмних та апаратних закладок та інших засобів подолання системи захисту після відновлення її функціонування.

Своєчасність передбачає запобіжний характер заходів забезпечення безпеки інформації, тобто постановку завдань щодо комплексного захисту в комп’ютерно-інтегрованих та робототехнічних системах та реалізацію заходів забезпечення безпеки інформації на ранніх стадіях розробки комп’ютерно-інтегрованих та робототехнічних систем загалом та їх системи захисту інформації, зокрема.

Розробка системи захисту повинна вестися паралельно з розробкою та розвитком самої системи, що захищається. Це дозволить врахувати вимоги безпеки при проектуванні архітектури і, зрештою, створити ефективніші (як за витратами ресурсів, так і за стійкістю) захищені системи.

Спадкоємність та вдосконалення передбачають постійне вдосконалення заходів та засобів захисту інформації на основі наступності організаційних та технічних рішень, кадрового складу, аналізу функціонування комп'ютерно-інтегрованих та робототехнічних систем та їх системи захисту з урахуванням змін у методах та засобах перехоплення інформації та впливу на компоненти комп'ютерно-інтегрованих та робототехнічних систем, нормативних вимог щодо захисту, досягнутого вітчизняного та зарубіжного досвіду у цій галузі.

Принцип поділу функцій вимагає, щоб жоден співробітник організації не мав повноважень, що дозволяють йому одноосібно здійснювати виконання критичних операцій. Усі такі операції мають бути поділені на частини, та їх виконання має бути доручено різним співробітникам. Крім того, необхідно вживати спеціальних заходів щодо недопущення змови та розмежування відповідальності між цими співробітниками.

Розумна достатність передбачає відповідність рівня витрат на забезпечення безпеки інформації цінності інформаційних ресурсів величині можливої шкоди від їх розголошення, втрати, витоку, знищення та спотворення. Заходи та засоби забезпечення безпеки інформаційних ресурсів, що використовуються, не повинні помітно погіршувати ергономічні показники роботи комп'ютерно-інтегрованих та робототехнічних систем, в яких ця інформація

циркулює. Зайві заходи безпеки, окрім економічної неефективності, призводять до втоми та роздратування персоналу.

Створити абсолютно непереборну систему захисту неможливо. Поки інформація знаходиться в обігу, заходи, що вживаються, можуть лише знизити ймовірність негативних впливів або збитки від них, але не виключити їх повністю.

При достатній кількості часу та коштів можна подолати будь-який захист. Тому є сенс розглядати певний прийнятний рівень забезпечення безпеки.

Високоефективна система захисту коштує дорого, використовує під час роботи істотну частину ресурсів комп'ютерної системи та може створювати відчутні додаткові незручності користувачам. Важливо правильно вибрати той достатній рівень захисту, коли витрати, ризик і розмір можливої шкоди були б прийнятними (аналіз ризику).

Персональна відповідальність передбачає покладання відповідальності за забезпечення безпеки інформації та системи її обробки кожного співробітника у межах його повноважень. Відповідно до цього принципу розподіл прав та обов'язків співробітників будується таким чином, щоб у разі будь-якого порушення коло винуватців було чітко відоме або зведене до мінімуму.

Мінімізація повноважень означає надання користувачам мінімальних прав доступу відповідно до виробничої потреби. Доступ до інформації повинен надаватися лише в тому випадку та обсязі, в якому це необхідно співробітнику для виконання його посадових обов'язків.

Взаємодія та співробітництво передбачає створення сприятливої атмосфери у колективах підрозділі. У такій обстановці співробітники повинні усвідомлено дотримуватися встановлених правил і сприяти діяльності підрозділів забезпечення безпеки інформації.

Вжиті заходи та встановлені засоби захисту, особливо в початковий період їх експлуатації, можуть забезпечувати надмірний, так і недостатній рівень захисту. Для забезпечення можливості варіювання рівнем захищеності, засоби захисту повинні мати певну **гнучкість**. Особливо важливою ця властивість є в тих випадках, коли установку засобів захисту необхідно здійснювати на систему, що вже працює, не порушуючи процесу її нормального функціонування. Крім того, зовнішні умови та вимоги з часом змінюються. У таких ситуаціях властивість гнучкості системи захисту позбавляє власників комп’ютерно-інтегрованих та робототехнічних систем необхідності прийняття кардинальних заходів щодо повної заміни засобів захисту на нові.

Суть **принципу відкритості** алгоритмів і механізмів захисту у тому, що захист має забезпечуватися лише з допомогою секретності структурної організації та алгоритмів функціонування її підсистем. Знання алгоритмів роботи системи захисту має давати можливості її подолання (навіть авторам). Це, однак, не означає, що інформація про конкретну систему захисту має бути загальнодоступною.

Простота застосування засобів захисту означає, що механізми захисту повинні бути інтуїтивно зрозумілі і прості у використанні. Застосування засобів захисту не повинно бути пов'язане зі знанням спеціальних мов або з виконанням дій, що вимагають значних додаткових трудовитрат при звичайній роботі зареєстрованих встановленим порядком користувачів, а також не повинно вимагати від користувача виконання малозрозумілих рутинних операцій (введення декількох паролів і імен і т.д.).

Наукова обґрунтованість та технічна реалізованість означають, що всі інформаційні технології, технічні та програмні засоби, засоби та заходи захисту інформації повинні бути реалізовані на сучасному рівні розвитку науки і техніки, науково обґрунтовані з точки зору досягнення заданого рівня безпеки інформації та повинні відповідати встановленим нормам та вимогам щодо безпеки інформації.

Спеціалізація та професіоналізм передбачають залучення до

розробки засобів та реалізації заходів захисту інформації спеціалізованих організацій, найбільш підготовлених до конкретного виду діяльності щодо забезпечення безпеки інформаційних ресурсів, які мають досвід практичної роботи та державні ліцензії на право надання послуг у цій галузі. Реалізація адміністративних заходів та експлуатація засобів захисту має здійснюватися професійно підготовленими співробітниками (фахівцями підрозділів забезпечення безпеки інформації).

Взаємодія та координація передбачають здійснення заходів забезпечення безпеки інформації на основі взаємодії всіх зацікавлених міністерств і відомств, підприємств і організацій при розробці та функціонуванні комп'ютерно-інтегрованих та робототехнічних систем та їх системи захисту інформації, підрозділів та фахівців спеціалізованих підприємств та організацій у галузі захисту інформації, залучених для розробки системи захисту інформації в комп'ютерно-інтегрованих та робототехнічних системах.

Обов'язковість контролю передбачає обов'язковість та своєчасність виявлення та припинення спроб порушення встановлених правил забезпечення безпеки інформації на основі використовуваних систем та засобів захисту інформації при вдосконаленні критеріїв та методів оцінки ефективності цих систем та засобів.

Контроль за діяльністю будь-якого користувача, кожного засобу захисту та щодо будь-якого об'єкта захисту повинен здійснюватися на основі застосування засобів оперативного контролю та реєстрації та повинен охоплювати як несанкціоновані, так і санкціоновані дії користувачів.

9.5. Основні механізми захисту в комп’ютерно-інтегрованих та робототехнічних системах

Для захисту комп'ютерних систем від неправомірного втручання у процеси їх функціонування та НСД до інформації використовуються такі основні методи захисту (захисні механізми):

- ідентифікація, аутентифікація (доказ автентичності) користувачів системи;
- розмежування доступу користувачів до ресурсів системи та авторизація (присвоєння повноважень) користувачам;
- реєстрація та оперативне сповіщення про події, що відбуваються в системі (аудит);
- криптографічне закриття даних, що зберігаються і передаються каналами зв'язку;
- контроль цілісності та автентичності (справжності та авторства) даних;

- виявлення та нейтралізація дій комп'ютерних вірусів;
- виявлення вразливостей (слабких місць) системи;
- ізоляція (захист периметра) комп'ютерних мереж (фільтрація трафіку, приховування внутрішньої структури та адресації, протидія атакам на внутрішні ресурси тощо);
- виявлення атак та оперативне реагування.
- резервне копіювання
- маскування.

Перелічені механізми захисту можуть застосовуватись у конкретних технічних засобах та системах захисту у різних комбінаціях та варіаціях. Найбільший ефект досягається за умови їх системного використання в комплексі з іншими видами заходів захисту.

Список літератури

1. Скаковська А.М., Радивоненко О.С., Шалда К.В. (2012), “Кластеризація зображень для їх компресії на основі компонентного аналізу”, Вісник Сумського державного університету, С. 32-36.

2. Гороховський С.С., Мороз А.В. (2021), “Сегментація зображень із використанням генетичних алгоритмів”, С. 52-55.

3. Волосюк Ю.В. (2014), “Аналіз алгоритмів кластеризації для задач інтелектуального аналізу даних”, Збірник наукових праць Військового інституту Київського національного університету імені Тараса Шевченка, С. 112-119

4. Tara S., Reddy B., Ramesh G., Sandeep K. (2014), “Various Image Segmentation Methods Based On Partial Differential Equation-A Survey”, International Conference on Computer & Communication Technologies Vol. 3, Pp. 183-186.

5. Jain A.K., Murty M.N., Flynn P.J. (1999), “Data clustering: a review”, ACM Computing Surveys, Vol. 31, No. 3, September 1999, pp. 264-323.6.Дікарєв О.В. Цифрова обробка сигналів. Учбово-методичний посібник. / Державний університет інформаційно-комунікаційних технологій. – Київ, 2009. 117 с.

7. Тихонов Ю.О., Савченко В.А., Котенко А.М., Зідан А.М. Лабораторний практикум з теорії кіл і сигналів в інформаційному та кіберпросторах. Практикум. - К.: ДУТ, 2019.- 221с.

8. Дікарев О.В. Навчально-методичний посібник по основам цифрової обробки сигналів. / Державний університет інформаційнокомунікаційних технологій. – Київ, 2014. 105 с.
9. Фриз М.Є., Стадник М. А Обробка сигналів та зображень / Конспект лекцій з дисципліни.- Тернопіль: ТНТУ, 2015. – 97 с.
10. Схемотехніка-1. Аналогова схемотехніка: Лабораторний практикум. [Електронний ресурс] : навч. посіб. для студ. спеціальності 171 «Електроніка»/ КПІ ім. Ігоря Сікорського; уклад.: Ю.О. Оникієнко, А.Ю. Мицукова. – Електронні текстові данні (1 файл: 3 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2021. –107 с.
11. Загальний огляд з аналізу даних "Data Science for Business" (Foster Provost, Tom Fawcett).

12. S.K. Mitra, 'Digital Signal Processing – A Computer Based Approach', 4th Edition, Tata McGraw Hill, New Delhi,. 2011.

13. Oppenheim, AV, RW Schafer, and JR Buck. Discrete-Time Signal Processing. 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1999. ISBN: 0137549202.

14. Proakis, J.G. and Manolakis, D.G. (2007) Digital Signal Processing: Principles, Algorithms, and Applications. 4th Edition, Pearson Education, Inc., New Delhi.

15. Digital filters / R.W. Hamming. Englewood Cliffs, New Jersey : Prentice Hall, 1989. xiv, 284 pages : illustrations ; 24 cm.

16. W. K. Chen. The Electrical Engineering Handbook. Publisher: Elsevier Science. January 2004. ISBN: 9780080477480

17. Ткач Є. Загальна теорія статистики : підруч. Київ : Центр навчальної літератури, 2017.

18. Основи цифрової обробки сигналів: https://www.tutorialspoint.com/digital_signal_processing/index.htm

19. Чернівецький національний університет імені Юрія Федьковича, Навчальний посібник «Основи та методи цифрової обробки сигналів: від теорії до практики», 2021

20. Комп'ютерний практикум «МАШИННЕ НАВЧАННЯ», Київ КПІ ім. Ігоря Сікорського 2022.

21. Уманець Т., Пігаєв Ю. Статистика : підруч. Київ : Вікар, 2013,

22. Навчальний посібник «Цифрова обробка сигналів та зображень» Київ КПІ ім. Ігоря Сікорського 2025

23. Hill T. STATISTICS: Methods and Applications / T. Hill, P.Lewicki. – Tulsa: StatSoft, OK, 2007. – 719 p. (StatSoft, Inc. (2011). Electronic Statistics Textbook. Tulsa, OK: StatSoft) (http://www.statsoft.com/textbook/)

24. Бутник О.М. Економіко-математичне моделювання перехідних процесів у соціально-економічних системах: [монографія] / О.М. Бутник. – Х.:Видавничий Дім „ІНЖЕК"; СПД Лібуркіна Л.М., 2004.

25. Бахрушин В.Є. Методи аналізу даних: навчальний посібник для студентів / В.Є. Бахрушин. – Запоріжжя: КПУ, 2011.
26. Єріна А.М. Статистичне моделювання та прогнозування / А.М. Єріна. – Київ: КНТЕУ, 2001.
27. О. В. ПЕРЕГУДА О. А. КАПУСТЯН О. Б. КУРИЛКО. Навчальний посібник «СТАТИСТИЧНА ОБРОБКА ДАНИХ», 2022.
28. «ПОПЕРЕДНІЙ АНАЛІЗ ДАНИХ. ОПИСОВА СТАТИСТИКА» (https://stud.com.ua/93299/statistika/poperedniy_analiz_danih_opisova_statistika#734)
29. Підручник по прогнозуванню даних "Forecasting: Principles and Practice" (Rob J Hyndman, George Athanasopoulos).
30. Wieder, P., & Nolte, H. (2022). Toward data lakes as central building blocks for data management and analysis. Frontiers in big Data, 5.
31. Hai, R., Miller, R., Jarke, M., & Quix, C. J. (2020). Data Integration and Metadata Management in Data Lakes (Doctoral dissertation, Ph. D. Dissertation. RWTH Aachen University. https://doi. org/10.18154/ RWTH-202008233).
32. Береза А.М. Основи створення інформаційних систем: Навчальний посібник / Береза А.М. — Київ : КНЕУ, 1998.
33. Єрьоміна Н.В. Проектування баз даних: Навчальний посібник / Єрьоміна Н.В. — Київ : КНЕУ, 1998.
34. Методи обробки зображень та компютерний зір : навч. посіб. / С.М. Вовк, В.В. Гнатушенко, М.В. Бондаренко. – Д. : ЛІРА, 2016. – 148 с.

35. Data mining : practical machine learning tools and techniques / Ian H. Witten, Eibe Frank.– 2nd ed. p. cm.– (Morgan Kaufmann series in data management systems) Includes bibliographical references and index.
36. Ониськів П.А., Литвиненко Я.В. (2019), “Аналіз методів сегментації зображень”, Матеріали IV Міжнародної науково-технічної конференції „Теоретичні та прикладні аспекти радіотехніки, приладобудування і комп’ютерних технологій“ присвячена 80-ти річчю з дня народження професора Я.І. Проця, С. 48-49.
37. Grady L. (2006), “Random walks of image segmentation”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Pp. 1768-1783.
38. Dzung L.P., Chenyang Xu., Jerry L.P. (2000), “Current Methods in Medical Image Segmentation”, Annual Review of Biomedical Engineering Vol. 2, Pp. 315-337.

Кіктєв Микола Олександрович

Коваль Валерій Вікторович

Ромащук Олександр Миколайович

ОБРОБКА ІНФОРМАЦІЇ В КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ ТА РОБОТОТЕХНІЧНИХ СИСТЕМАХ

Конспект лекцій
Частина 2

Підписано до видання 11.11.2025 м.Київ. друк. арк. - 150.