

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**  
**МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА**

**Н. Д. Сізова**

**ІНТЕЛЕКТУАЛЬНІ УПРАВЛЯЮЧІ СИСТЕМИ І ТЕХНОЛОГІЇ**

**КОНСПЕКТ ЛЕКЦІЙ**

*(для здобувачів другого (магістерського) рівня вищої освіти денної, заочної і  
дистанційної форм навчання  
зі спеціальності 122 – Комп'ютерні науки)*

**Харків**  
**ХНУМГ ім. О. М. Бекетова**  
**2024**

УДК 004.891

**Сізова Н. Д.** Інтелектуальні управляючі системи і технології : конспект лекцій для здобувачів другого (магістерського) рівня вищої освіти денної, заочної і дистанційної форм навчання зі спеціальності 122 – Комп’ютерні науки / Н. Д. Сізова ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2024.– 65 с.

Автор

д-р фіз.-мат. наук, проф. Н. Д. Сізова

Рецензент

**М. Ю. Карпенко**, кандидат технічних наук, доцент кафедри комп’ютерних наук та інформаційних технологій (Харківський національний університет міського господарства імені О. М. Бекетова)

*Рекомендовано кафедрою комп’ютерних наук та інформаційних технологій, протокол № 1 від 29 серпня 2023 р.*

© Н. Д. Сізова, 2024

© ХНУМГ ім. О. М. Бекетова, 2024

## ЗМІСТ

ВСТУП.....	5
ТЕМА 1 ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ СИСТЕМИ .....	6
1.1 Поява і застосування інтелектуальних інформаційних технологій.....	6
1.2 Види інтелектуальних систем.....	8
1.3 Класифікація завдань, вирішуваних інтелектуальними інформаційними системами.....	9
ТЕМА 2 РОЗРОБКА І ПРОЄКТУВАННЯ ІНТЕЛЕКТУАЛЬНИХ ІНФОРМАЦІЙНИХ СИСТЕМ .....	11
2.1 Етапи проєктування інтелектуальних інформаційних систем.....	11
2.2 Стадії існування інтелектуальних інформаційних систем.....	12
2.3 Інструментальні засоби проєктування інтелектуальних інформаційних систем.....	13
2.4 Архітектура інтелектуальних інформаційних систем.....	14
2.5 Інтелектуальні інформаційні системи в управлінні знаннями.....	15
ТЕМА 3 ПОДАННЯ ЗНАНЬ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ .....	17
3.1 Підходи до подання знань.....	17
3.2 Дані та знання.....	18
3.3 Властивості та моделі знань.....	21
3.4 Неоднорідність знань. Галузі та рівні знань.....	22
3.5 База знань як об'єднання простіших одиниць.....	22
3.6 Моделі подання знань в інтелектуальних інформаційних системах.....	25
ТЕМА 4 КЕРУВАННЯ ІНТЕЛЕКТУАЛЬНИМИ СИСТЕМАМИ.....	26
4.1 Алгоритмічний та декларативний підходи до керування.....	26
4.2 Формалізація понять алгоритмічності та декларативності.....	27
4.3 Характеристика інтелектуальних систем з погляду кібернетики.....	30
ТЕМА 5 МЕТОДИ І СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ .....	31
5.1 Основні положення теорії штучного інтелекту.....	31
5.2 Системи розпізнавання образів.....	32
5.3 Нейрокомп'ютер – програмно-технічна система.....	33
5.4 Модель штучного нейрона.....	34
5.5 Розпізнавання образів. Основні поняття. Математичні моделі.....	38
ТЕМА 6 ОСНОВИ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ.....	41
6.1 Проблематика нейронних мереж і генетичних алгоритмів.....	41
6.2 Модель штучного нейрона.....	42

6.3 Архітектура з'єднань штучних нейронів.....	44
6.4 Навчання штучної нейронної мережі.....	47
6.5 Мережі Кохонена і Хопфілда.....	50
<b>ТЕМА 7 ЕКСПЕРТНІ СИСТЕМИ – СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ .....</b>	<b>53</b>
7.1 Основа експертної системи.....	53
7.2 Технологія побудови експертної системи.....	54
7.3 Структура експертної системи.....	56
<b>СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ .....</b>	<b>64</b>

## ВСТУП

Характерною властивістю сьогодення є процеси інформатизації, що інтенсивно розвиваються та впроваджуються практично у всіх сферах людської діяльності. Вони спричинили формування нової інформаційної інфраструктури, яка пов'язана з новим типом суспільних відносин та використанням нових інформаційних технологій.

Одним із напрямків розвитку штучного інтелекту є розробка комп'ютерних інтелектуальних систем, здатних виконувати функції, що традиційно вважаються інтелектуальними – розуміння мови, логічний висновок, використання накопичених знань, навчання, розпізнавання образів, а також навчатися і пояснювати свої рішення. На сьогодні інформаційні інтелектуальні системи є перспективними у своєму розвитку.

У конспекті розглядаються питання, присвячені інтелектуальним інформаційним системам і технологіям, організації, проектування, розробки і застосування систем для обробки інформації, які базуються на використанні методів штучного інтелекту.

У матеріалі подаються сім тем, список рекомендованої літератури. Розглядаються фундаментальні питання штучного інтелекту, поняття та класифікація інтелектуальних інформаційних систем. Міститься інформація про моделі подання знань, зокрема логічні, фреймові, семантичні моделі, онтології. Розглядаються експертні системи та нейронні мережі.

До кожного розділу наведені контрольні запитання для самоконтролю, що дозволить студентам перевірити свої знання за теоретичними темами.

Це видання призначено для здобувачів вищої освіти спеціальностей 122 – Комп'ютерні науки, а також може бути корисним під час вивчення сучасних інтелектуальних інформаційних технологій.

# ТЕМА 1 ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ СИСТЕМИ

## 1.1 Поява і застосування інтелектуальних інформаційних технологій

*Інтелектуальна інформаційна система* (далі – ІС) – це один з видів автоматизованих інформаційних систем, інколи ІС називають системою, заснованою на знаннях. ІС є комплексом програмних, лінгвістичних і логіко-математичних засобів для реалізації основного завдання: здійснення підтримки діяльності людини і пошуку інформації в режимі розширеного діалогу на природній мові.

Основне призначення інформаційних систем – це своєчасне подання необхідної інформації особі, що приймає рішення (далі – ОПР) для ухвалення адекватних і ефективних рішень під час керування процесами, ресурсами, фінансовими транзакціями, персоналом або організацією загалом. Однак у процесі розвитку інформаційних технологій, дослідження операцій і технологій моделювання, а також зі зростанням споживачів інформаційно-аналітичної підтримки самих ОПР, усе більше проявлялася потреба в системах, що не тільки представляють інформацію, але й здійснюють деякий її попередній аналіз, здатних давати деякі ради й рекомендації, здійснювати прогнозування розвитку ситуацій, відбирати найбільш перспективні альтернативи рішень, тобто підтримувати рішення ОПР, взявши на себе значну частину рутинних операцій, а також функції попереднього аналізу й оцінок [1–2].

Інформаційна система підтримки рішень (далі – ІСПР) зв'язує інтелектуальні ресурси ОПР (особи, що приймає рішення) зі здатностями й можливостями комп'ютера для поліпшення якості рішень. Ці системи призначені для ухвалення рішень в умовах напівструктурованих і слабо структурованих завдань.

Інтелектуальні інформаційні технології (далі – ІІТ) (Intellectual information technology, ІІТ) – це інформаційні технології, що допомагають людині прискорити аналіз політичної, економічної, соціальної й технічної ситуації, а також синтезувати управлінські рішення.

Використання ІІТ у реальній практиці має на увазі облік специфіки проблемної області, що може характеризуватися таким набором ознак:

- якість і оперативність ухвалення рішень;
- нечіткість цілей і інституціональних границь;
- множинність суб'єктів, що беруть участь у вирішенні проблеми;
- хаотичність, флуктуація і квантованість поведінки середовища;
- множинність взаємного впливу один на одного факторів;
- слабка формалізованість, унікальність, не стереотипність ситуацій;
- латентність, прихованість, наявність інформації;
- девіантність реалізації планів, значущість малих дій;
- парадоксальність логіки рішень і ін.

ІІТ формуються під час створення інформаційних систем і інформаційних технологій для підвищення ефективності керування знаннями, ухвалення рішень в умовах, пов'язаних із виникненням проблемних ситуацій. У цьому випадку

будь-яка життєва або ділова ситуація описується у вигляді деякої пізнавальної моделі (когнітивної схеми, архетипу, фрейму та ін.), що згодом використовується як підстава для побудови й проведення моделювання, зокрема комп'ютерного [3–4].

Історія інтелектуальних інформаційних систем (ІС) починається із середини ХХ століття, що пов'язане з розвитком штучного інтелекту як нового наукового напрямку, появою терміна «Artificial Intelligence».

З середини 1970-х років починається практичне застосування інтелектуальних інформаційних систем, систем штучного інтелекту, систем підтримки рішень і інформаційних систем.

Історія ІТ зв'язана також із розвитком трьох наукових напрямків: комп'ютерної філософії, комп'ютерної психології й просунутої комп'ютерної науки (Advanced computer science) і доповнюється прогресом у створенні:

- 1) ситуаційних центрів;
- 2) інформаційно-аналітичних систем;
- 3) інструментаріїв еволюційних обчислень і генетичних алгоритмів;
- 4) систем підтримки спілкування людини з комп'ютером природною мовою;
- 5) когнітивним моделюванням;
- 6) систем автоматичного тематичного рубрикування документів;
- 7) систем стратегічного планування;
- 8) інструментаріїв технічного й фундаментального аналізу фінансових ринків;
- 9) систем менеджменту якості;
- 10) систем керування інтелектуальною власністю й ін.

Ключові події в історії ІС й інженерії знань почалися з першої роботи У. Маккалока й У. Питса в 1943 р. [4], у якій подається логічне вирахування ідей, властивих нервової діяльності.

Розробки в області ІС велися у двох основних напрямках:

- перший напрямок пов'язаний зі спробами розробки інтелектуальних машин шляхом моделювання їхнього біологічного прототипу – людського мозку.

Зараз цей напрямок відроджується на основі розвитку сучасних апаратних і програмних засобів (мікročіпи на основі нечіткої логіки, розподілені багатопроесорні системи, багатоагентні системи, м'які обчислення, генетичні алгоритми й нейронні мережі тощо);

- другий напрямок пов'язаний з розробками методів, прийомів, спеціалізованих пристроїв і програм для комп'ютерів, що забезпечують рішення складних математичних і логічних завдань, що дозволяють автоматизувати окремі інтелектуальні дії людини (системи, засновані на знаннях, експертні системи, прикладні інтелектуальні системи).

Ці два напрямки визначають програму мінімум і програму максимум, між якими й лежить галузь сьгоднішніх досліджень і розробок систем ІІ. Роботи з розробки програмного й апаратного забезпечення ІІ виділені в окрему галузь.

## **Інтелектуальні системи**

Інтелектуальна система (ІС, intelligent system) – це технічна або програмна система, здатна вирішувати завдання, що традиційно вважаються творчими, приналежні конкретної предметної області, знання про яку зберігаються в пам'яті такої системи.

Структура інтелектуальної системи містить три основних блоки – базу знань, вирішувач і інтелектуальний інтерфейс.

З усім процесом розробки інтелектуальних інформаційних систем у цілому й ІС, зокрема, тісно зв'язана інженерія знань. Це методологія ІС, що охоплює методи видобутку, аналізу й вираження в правилах знань експертів для формування бази правил. Розвиток ІС створив інженерію знань – процес побудови інтелектуальних систем. Вона становить сукупність моделей, методів і технічних прийомів, націлених на створення систем, які призначені для вирішення проблем з використанням знань. Головними елементами інженерії знань є використання операцій типу узагальнення, генерація гіпотез для індуктивних висновків, підготовка нових програм самими комп'ютерними програмами тощо.

### **1.2 Види інтелектуальних систем**

#### **1. Розрахунково-логічна система.**

До розрахунково-логічних систем відносять системи, здатні вирішувати управлінські й проєктні завдання по декларативних описах умов. Разом із тим користувач має можливість контролювати в режимі діалогу всі стадії обчислювального процесу. Ці системи здатні автоматично будувати математичну модель завдання й автоматично синтезувати обчислювальні алгоритми із формулювання завдання. Ці властивості реалізуються завдяки наявності бази знань у вигляді функціональної семантичної мережі й компонентів дедуктивного виводу й планування.

#### **2. Рефлекторна інтелектуальна система.**

Рефлекторна система – це система, що формує створені спеціальними алгоритмами відповідні реакції на різні комбінації вхідних впливів. Алгоритм забезпечує вибір найбільш імовірної реакції інтелектуальної системи на безліч вхідних впливів, за відомих ймовірностей вибору реакції на кожний вхідний вплив, а також на деякі комбінації вхідних впливів.

Це завдання подібне тим, що реалізують перцептрони. Перцептрон, або персептрон (perceptron) – математична й комп'ютерна модель сприйняття інформації мозком (кібернетична модель мозку), запропонована Фрэнком Розенблаттом в 1957 р. і реалізована у вигляді електронної машини «Марко-1» в 1960 р. Перцептрон став однією з перших моделей неймереж, «Марко-1» – першим у світі нейрокомп'ютером. Незважаючи на свою простоту, перцептрон здатний навчатися й вирішувати достатньо складні завдання. Рефлекторні програмні системи застосовуються до таких завдань: природно-мовний доступ до баз даних; оцінки інвестиційних пропозицій; оцінки й прогнозування впливу шкідливих речовин на здоров'я населення; прогнозування результатів спортивних ігор [3].



### 3. Інтелектуальна інформаційна система.

Інтелектуальна інформаційна система (ІС, intelligent system) – система, заснована на знаннях.

### 4. Гібридна інтелектуальна система.

Під гібридною інтелектуальною системою (далі – ГІС) прийнято розуміти систему, у якій для вирішення завдання використовується більше одного методу імітації інтелектуальної діяльності людини.

У такий спосіб ГІС – це сукупність:

- аналітичних моделей;
- експертних систем;
- штучних нейронних мереж;
- нечітких систем;
- генетичних алгоритмів;
- імітаційних статистичних моделей.

**Інтелектуальні** інформаційні системи (ІС) підтримки ухвалення рішень становлять комплекс програмних, лінгвістичних і логіко-математичних засобів для реалізації основного завдання: здійснення підтримки діяльності людини, наприклад можливість пошуку інформації в режимі просунутого діалогу природною мовою.

ІС – це комп'ютерна система, що складається з п'яти основних взаємодіючих компонентів: мовної підсистеми (механізм забезпечення зв'язку між користувачем і іншими компонентами ІСПР), інформацією підсистеми (сховище даних і засобів їхньої обробки), підсистеми керування знаннями (сховище знань про проблемну область, таких як процедури, евристики й правила, і засобу обробки знань), підсистеми керування моделями й підсистеми обробки й рішення завдань (сполучна ланка між іншими підсистемами).

### **1.3 Класифікація завдань, вирішуваних інтелектуальними інформаційними системами**

Розглянемо завдання, які вирішують інтелектуальні інформаційні системи.

- **Інтерпретація даних.** Це одне із традиційних завдань для експертних систем. Під інтерпретацією розуміється процес визначення змісту даних, результати якого повинні бути погодженими й коректними. Зазвичай передбачається різноманітний аналіз даних.

- **Діагностика.** Під діагностикою розуміється процес співвідношення об'єкта з деяким класом об'єктів і/або виявлення несправності в деякій системі. Несправність – це відхилення від норми. Таке трактування дозволяє з єдиних теоретичних позицій розглядати й несправність устаткування в технічних системах, і захворювання живих організмів, і всілякі природні аномалії. Важливою специфікою є тут необхідність розуміння функціональної структури («анатомії») діагностуючої системи.

- **Моніторинг.** Основне завдання моніторингу – безперервна інтерпретація даних у реальному масштабі часу й сигналізація про вихід тих або інших параметрів за припустимі межі. Головні проблеми – «пропуск» тривожної

ситуації й інверсне завдання «помилкового» спрацьовування. Складність цих проблем у розмитості симптомів тривожних ситуацій і необхідність обліку тимчасового контексту.

- **Проектування.** Проектування складається в підготовці специфікацій на створення «об'єктів» із заздалегідь певними властивостями. Під специфікацією розуміється весь набір необхідних документів-креслення, пояснювальна записка й т. д. Основні проблеми тут – одержання чіткого структурного опису знань про об'єкт і проблема «сліду». Для організації ефективного проектування або перепроєктування необхідно формувати не тільки самі проектні рішення, але й мотиви їхнього прийняття. Таким чином, у завданнях проектування тісно зв'язуються два основних процеси, виконуваних у межах відповідної ІС: процес виводу рішення й процес пояснення.

- **Прогнозування.** Прогнозування дозволяє пророкувати наслідки деяких подій або явищ на підставі аналізу наявних даних. Прогнозуючі системи логічно виводять імовірні наслідки із заданих ситуацій. У прогнозуючій системі зазвичай використовується параметрична динамічна модель, у якій значення параметрів «підганяються» під задану ситуацію. Виведені із цієї моделі наслідку становлять основу для прогнозів з імовірнісними оцінками.

- **Планування.** Під плануванням розуміється знаходження планів дій, що ставляться до об'єктів, здатним виконувати деякі функції. У таких ІС використовуються моделі поведінки реальних об'єктів для того, щоб логічно вивести наслідки планованої діяльності.

- **Навчання.** Під навчанням розуміється використання комп'ютера для навчання якійсь дисципліні або предмету. Системи навчання діагностують помилки під час вивчення якої-небудь дисципліни за допомогою ЕОМ і підказують правильні рішення. Вони акумулюють знання про гіпотетичного «учня» і його характерні помилки, потім у роботі вони здатні діагностувати слабкості в пізнаннях тих, яких навчають, і знаходити відповідні засоби для їхньої ліквідації. Крім того, вони планують спілкування з учнем залежно від успіхів учня з метою передачі знань.

Нейронні мережі не програмуються у звичному змісті цього слова, вони навчаються. Можливість навчання – одне з головних переваг нейронних мереж перед традиційними алгоритмами. Технічно навчання полягає в знаходженні коефіцієнтів зв'язків між нейронами. У процесі навчання нейронна мережа здатна виявляти складні залежності між вхідними даними й вихідними, а також виконувати узагальнення. Це значить, що, у випадку успішного навчання, мережа зможе повернути правильний результат на підставі даних, які були відсутні в навчальній вибірці.

- **Керування.** Під керуванням розуміється функція організованої системи, що підтримує певний режим діяльності. Такого роду ІС здійснюють керування поведінкою складних систем відповідно до заданих специфікацій.

- **Підтримка прийняття рішень.** Підтримка ухвалення рішення – це сукупність процедур, що забезпечує особу, що приймає рішення, необхідною інформацією й рекомендаціями, які полегшують процес ухвалення рішення. Ці

ЕС допомагають фахівцям вибрати й/або сформувані потрібну альтернативу серед безлічі виборів під час ухвалення відповідальних рішень.

У загальному випадку всі системи, засновані на знаннях, можна підрозділити на системи, що вирішують завдання аналізу, і на системи, що вирішують завдання синтезу. Основна відмінність завдань аналізу від завдань синтезу полягає в тім, що, якщо в завданнях аналізу, безліч рішень може бути перераховане й включено в систему, то в завданнях синтезу безліч рішень потенційно не обмежене й будується з рішень компонентів або підпроблем. Завданнями аналізу є: інтерпретація даних, діагностика, підтримка ухвалення рішення; до завдань синтезу належать проектування, планування, керування. Комбіновані: навчання, моніторинг, прогнозування.

### **Контрольні запитання**

1. Надайте визначення інтелектуальних інформаційних систем.
2. Назвіть види інтелектуальних інформаційних систем.
3. Опишіть шляхи розвитку інтелектуальних інформаційних систем.
4. Що є основою інтелектуальних інформаційних систем?

## **ТЕМА 2 РОЗРОБКА І ПРОЄКТУВАННЯ ІНТЕЛЕКТУАЛЬНИХ ІНФОРМАЦІЙНИХ СИСТЕМ**

### **2.1 Етапи проектування інтелектуальних інформаційних систем**

Існують різні погляди на визначення кількості етапів проектування інтелектуальних систем. Це залежить від багатьох факторів, зокрема від характеру функцій майбутньої інтелектуальної системи, галузі використання, наявності розвинених інструментальних засобів тощо.

Процес побудови систем II можна розділити на п'ять етапів (рис. 2.1).

Ідентифікація визначення завдань і ідентифікація їхніх характеристик. Розробляється технічне завдання на проєктовану систему, обмежується коло користувачів системи.

Виділення головних концепцій предметної області, які відбивають знання кола експертів. Інженер знань визначає формальні засоби подання знань і процедури одержання рішень. Виявляються й формулюються поняття, що визначають вибір характерної схеми подання знань експерта про предметну область. Основним джерелом знань про проблемну область є людина-експерт, книги, технологічні описи, інструкції, документи, методи «мозкового штурму», методи автоматизованого заповнення бази знань. Іншим важливим джерелом знань є мережа «Інтернет» (традиційний пошук необхідної інформації й знань, а також інтелектуальні агенти (програмні роботи)) [3–4].

На рисунку 2.1 подані основні етапи проектування інтелектуальних інформаційних систем [3].

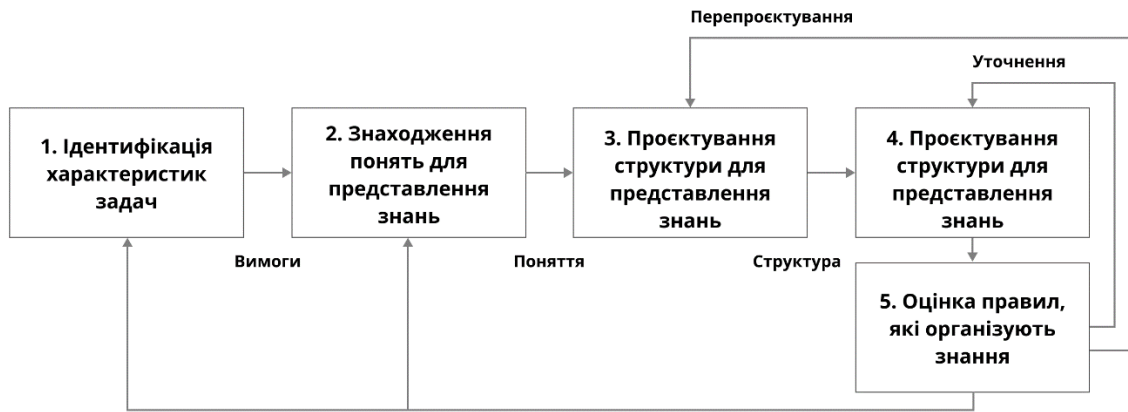


Рисунок 2.1– Етапи проєктування ІС

Вибір формалізму подання знань і визначення механізму виводу рішень. Розроблена структура для подання знань є основою для реалізації наступного етапу – безпосередньої побудови бази знань системи.

Вибір або розробка мови подання знань. Після того як правила сформульовані й представлені обраними мовою подання, вони заносяться інженером знань у бази знань.

Тестування системи шляхом рішення конкретних перевірочних завдань.

Етапи створення інтелектуальних систем не є чітко обкресленими й докладно регламентованими. Між деякими з них важко провести тимчасову й змістовну границю. Вони в якомусь ступені приблизно описують процес проєктування інтелектуальних систем.

## 2.2 Стадії існування інтелектуальних інформаційних систем

Стадії існування інтелектуальних систем (або життєві цикли системи) відповідають рівню готовності системи, завершеності її функціональних можливостей, реалізованих інструментарієм. Визначають такі стадії існування інтелектуальних систем: демонстраційний прототип; дослідницький прототип; діючий прототип; промислова система; комерційна система.

Демонстраційний прототип – це стан розробленості системи, коли вона вирішує частину проблемних завдань. При розробці демонстраційного прототипу прагнуть досягти суперечливих цілей: з одного боку, система на стадії демонстраційного прототипу повинна виконувати завдання, які б досить повно характеризували її можливості, з іншого боку, цю стадію прагнуть пройти як найшвидше. Робота демонстраційного прототипу може бути визнана задовільною, якщо він оперує мінімальним набором правил, достатнім для рішення деяких завдань. Час розробки коливається від двох місяців до року.

Дослідницький прототип проєктується протягом 1,5–2 років. На цій стадії розвитку системи її база знань уже містить кілька сотень правил, які достатньо адекватно описують предметну область.

Діючий прототип інтелектуальних систем здійснює якісний вивід рішень на просторі, що розширився, правил, що достигли порядку 1 000. Тому для виводу складних рішень потрібні більші ресурси часу й пам'яті.

Промислові системи забезпечують високий рівень якості рішення проблем предметної області при значних зменшеннях часу рішення й необхідної пам'яті. Кількість правил зростає не настільки значно порівняно з діючим прототипом. На цій стадії відбувається перетворення діючого прототипу шляхом розширення кількості правил і вдосконалювання інтелектуальних систем на базі використання більш ефективних, інструментальних засобів. Це вимагає приблизно 3–4 роки.

Комерційна система призначена переважно для продажу. Вона є або проблемно-орієнтованою, або проблемно-незалежною.

### **2.3 Інструментальні засоби проєктування інтелектуальних інформаційних систем**

Незважаючи на наявність безлічі підходів як до розуміння завдань ІС, так і створення інтелектуальних інформаційних систем, можна виділити два основних підходи до розробки ІС:

- спадний ( Top-Down AI), семіотичний – створення експертних систем, баз знань і систем логічного виводу, що імітують високо рівневі психічні процеси: мислення, міркування, мова, емоції, творчість тощо;
- висхідний ( Bottom-Up AI), біологічний – вивчення нейронних мереж і еволюційних обчислень, що моделюють інтелектуальне поведіння на основі біологічних елементів, а також створення відповідних обчислювальних систем, таких як нейрокомп'ютер або біокомп'ютер.

Для розробки ІС раніше використовувалися логічні мови (Пролог, ЛПП і т. д.), а зараз використовуються різні процедурні мови.

Логіко-математичне забезпечення розробляється як для самих модулів систем, так і для стикування цих модулів. Однак в галузі лінгвістики теж існує безліч проблем, наприклад, для забезпечення роботи системи в режимі діалогу з користувачем природною мовою необхідно закласти в систему алгоритми формалізації природної мови, а це завдання виявилось куди більше складним, ніж передбачалося на зорі розвитку інтелектуальних систем. Ще одна проблема – постійна мінливість мови, що обов'язково має бути відбита в системах штучного інтелекту.

На проєктування й створення однієї експертної системи раніше було потрібно 20–30 людини-років. Зараз є низка засобів, що прискорюють створення. Ці засоби називають інструментальними або інструментарієм. Використання інструментальних засобів розробки експертних систем скорочує час на їхнє створення в 3–5 разів.

Інструментальний засіб розробки експертних систем – це мова програмування, яка використовується інженером знань і/або програмістом для побудови експертної системи. Цей інструмент відрізняється від звичайних мов програмування тим, що забезпечує зручні способи подання складних високорівневих понять.

За своїм призначенням й функціональними можливостями інструментальні програми, які застосовуються під час проєктування експертних систем, можна розділити на чотири категорії.

### 1. Оболонки експертних систем.

Системи цього типу створюються зазвичай на основі якої-небудь експертної системи, що достатньо добре зарекомендувала себе на практиці. При створенні оболонки із системи-прототипу видаляються компоненти, занадто специфічні для області її безпосереднього застосування, і залишаються ті, які не мають вузької спеціалізації. Прикладом може бути система ЕМУСІН, створена на основі минулої тривалої «обкатки» системи МУСІН. В ЕМУСІН збережений інтерпретатор і всі базові структури даних і таблиці знань, які пов'язані з ними механізми індексації. Оболонка доповнена спеціальною мовою, що поліпшує читабельність програм, і засобами підтримки бібліотеки типових випадків і висновків, виконаних експертною системою.

### 2. Мови програмування високого рівня.

Інструментальні засоби цієї категорії дають можливість розробнику не заглиблюватися в деталі реалізації системи – способи ефективного розподілу пам'яті, рівневі процедури доступу й маніпулювання даними. Однією з найбільш відомих представників таких мов є OPS5. Ця мова проста у вивченні й надає програмістові набагато ширші можливості, ніж типові спеціалізовані оболонки. Варто зазначити, що більшість подібних мов так і не було доведено до рівня комерційного продукту і становлять скоріше інструмент для дослідників. Здійснюється програмування на звичайних мовах (Паскаль, Сі й 14н.), програмування на спеціалізованих мовах, які застосовуються у завданнях штучного інтелекту (LISP, FRL, SmallTalk ) і ін.

### 3. Середовище програмування, що підтримує кілька парадигм.

Засоби цієї категорії включають кілька програмних модулів, що дозволяє користувачеві комбінувати в процесі розробки експертної системи різні стилі програмування. Серед перших проєктів такого роду була дослідницька програма LOOP, що допускала використання двох типів подання знань: базованого на системі правил і об'єктно-орієнтованого. На основі цієї архітектури в другій половині 1980-х років було розроблено кілька комерційних програмних продуктів, з яких найбільшу популярність одержали KEE, KnowledgeCraft і ART. Ці програми надають у розпорядження кваліфікованого користувача безліч опцій і для наступних розробок, таких як KAPPA й CLIPS, і стали свого роду стандартом. Однак освоїти ці мови програмістам далеко не так просто, як мови, віднесені до попередньої категорії.

### 4. Додаткові модулі.

Засоби цієї категорії становлять автономні програмні модулі, призначені для виконання специфічних завдань у межах обраної архітектури системи вирішення проблем.

## **2.4 Архітектура інтелектуальних інформаційних систем**

Розглянемо структуру інтелектуальної системи.

Залежно від характеру виконуваних функцій і області дій експерти виконують кілька характерних завдань, які є типовими: інтерпретація, планування, керування, проєктування, прогнозування, диспетчеризація й моніторинг, діагностика. Головне, експерт повинен оновляти свої знання (тобто

навчатися), пояснювати дії, обґрунтовувати рішення, прогнозувати розвиток ситуацій, активно взаємодіяти із зовнішнім середовищем і сприймати інформацію різного характеру, одержувати рішення на основі наявних знань, зберігати в пам'яті необхідну інформацію й фактографічні дані. Аналіз завдань слугує орієнтиром при розгляді архітектури ІС, заснованих на знаннях.

Таким чином, щоб створити систему, що працює зі знаннями й здатну якоюсь мірою замінити експерта або допомогти йому в ухваленні рішень під час керування виробництвом, необхідно прагнути закласти в архітектуру системи можливості із реалізації названих функцій.

На рисунку 2.2 подана узагальнена структура й компоненти інтелектуальної системи, а також її оточення [3].



Рисунок 2.2 – Структура інтелектуальних систем

### 2.5 Інтелектуальні інформаційні системи в управлінні знаннями

Тенденції розвитку інтелектуальних інформаційних систем у керуванні знаннями характеризуються основними напрямками, які пов'язані з розробкою моделей і методів реалізації окремих аспектів одержання й перетворення знань.

1. Технології добування й подання знань. У першому випадку основним завданням є розробка методів: формального опису «ознак знань» (пошукових образів); формалізації процесу (Далі – Про); розпізнавання й порівняння образів; добування знань із експертів, статистики, текстів, «досвіду» і т. п. У другому – вирішуються завдання, пов'язані з формалізацією знань для їхнього подання в пам'яті інтелектуальних систем (ІС). Рішення цих завдань дозволяє розроблювачам комплексних технологій одержати відповіді на три принципово важливих питання: які знання необхідно подавати в ІС, хто (що) є джерелом цих знань, які методи й моделі забезпечують адекватне подання цих знань в ІС.

2. Технології маніпулювання знаннями, рішення інтелектуальних завдань припускає не тільки подання знань в ІС, але і їхню обробку, тобто необхідно навчити ІС оперувати ними. Тому тут вивчаються питання поповнення знань на основі їхніх неповних описів, класифікації знань в ІС, розробляються процедури й методи узагальнення знань, достовірного виводу й ін.

3. Технологія спілкування. Перехід до ІС знаменує нову технологію спілкування кінцевих користувачів з ЕОМ і вимагає рішення таких проблем, як розуміння зв'язних текстів на обмеженій і необмеженій природній мові, розуміння мови і її синтез, розробка комунікативних моделей «користувач-ЕОМ», формування пояснень і т. п. Головна мета даних досліджень – забезпечення комфортних умов для спілкування людини й ІС.

4. Технології сприйняття. Розробка цих технологій припускає створення методів аналізу тривимірних сцен, подання інформації про зорові образи в базі знань ІС, трансформації зорових сцен у текстові описи й назад, а також розробку процедур когнітивної графіки й ін.

5. Технології навчання. Відмінною рисою ІС є здатність вирішувати завдання, у явному вигляді не подані в базі знань, що вимагає наділення ІС здатністю до навчання. Для цих цілей необхідно створити методи формування умов завдання по опису проблемної ситуації або за спостереженням за цією ситуацією, забезпечити перехід від відомого рішення часткових завдань до рішення загального завдання, наділити ІС здатністю декомпонувати вихідне завдання на більш дрібні, рішення яких відомо, розробити нормативні й декларативні моделі самого процесу навчання, створити теорію наслідувального поведіння й ін.

6. Технології поведіння. Взаємодія ІС із середовищем вимагає розробки спеціальних поведінкових процедур, які б дозволили їм адекватно реагувати на ті або інші зміни в середовищі. Така взаємодія припускає створення моделей доцільного, нормативного й ситуативного поведіння, а також розробку методів багаторівневого планування й корекції планів у динамічних ситуаціях.

Галузь застосування існуючих на сьогоднішній день систем ІІ охоплює безліч сфер: медичну діагностику, інтерпретацію геологічних даних, наукові дослідження в хімії й біології, військова справа, виробництво, фінанси й інші області. Однак, незважаючи на значні успіхи в галузі ІІ, поки ще існує певний розрив між технічними розробками, програмними засобами ІІ й можливостями їх ширшого практичного застосування, зокрема в економіці.

Найпоказовішим сектором, що акумулює різні проблемні напрямки економічної галузі, є керування промисловим підприємством. На його прикладі особливо добре видні переваги використання систем ІІ для вирішення як різних предметних завдань, так і для керування інтегрованою системою підприємства загалом.

Існує безліч доводів на користь того, що системи штучного інтелекту можуть і повинні стати найважливішою складовою в технології сучасних виробництв. Основними з них є:



- подолання складності (складності керування виникають тоді, коли доводиться робити вибір з множини можливих рішень);
- керування підприємством вимагає організації більших обсягів інформації;
- як зменшити інформацію до того рівня, що необхідний для ухвалення рішення (втрата інформації, що надходить від об'єктів, що працюють у реальному режимі часу, може істотно позначитися на результаті);
- нестача часу на ухвалення рішення (проявляється в міру ускладнення виробництва);
- проблема координації (рішення необхідно координувати з іншими ланками процесу або об'єкта);
- необхідність збереження й поширення знань дуже досвідчених експертів, отриманих ними в процесі багаторічної роботи й великого практичного досвіду.

Проблема добування знань і їхні збереження й розподіли – сьогодні одна з головних проблем організацій.

Таким чином, інтелектуалізація інформаційних систем керування й трансформація їх в інтелектуальні інформаційні системи керування знаннями, підтримки ухвалення рішень є найбільш значущим і важливим напрямком.

### **Контрольні запитання**

1. Назвіть етапи розробки інтелектуальних інформаційних систем.
2. Які інструментальні засоби використовуються для проєктування інтелектуальних інформаційних систем?
3. Опишіть життєвий цикл інтелектуальних інформаційних систем.
4. Проаналізуйте основні елементи у структурі інтелектуальних інформаційних систем.
5. Які етапи керування знаннями інтелектуальних інформаційних систем.
6. Назвіть галузі застосування існуючих систем II.

## **ТЕМА 3 ПОДАННЯ ЗНАНЬ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ**

### **3.1 Підходи до подання знань**

**Знання** є інформаційною основою інтелектуальних систем, оскільки саме вони завжди зіставляють зовнішню ситуацію зі своїми знаннями і керуються ними під час ухвалення рішень. Не менш важливим є те, що знання – це систематизована інформація, яка може у певний спосіб поповнюватися і на основі якої можна отримувати нову інформацію, тобто нові знання [3–5].

Існує багато підходів до визначення поняття «знання». На сучасному етапі домінуючою парадигмою, що лежить в основі моделей подання знань у системах штучного інтелекту, можна вважати парадигму (певну сукупність ключових принципів), характерну для **символьного підходу**. Цю парадигму можна

охарактеризувати як **вербально-дедуктивну** або **словесно-логічну** через певні чинники:

- будь-яка інформаційна одиниця задається вербально, тобто у формі, наближеній до словесної, у вигляді набору явно сформульованих тверджень або фактів;
- основним механізмом отримання нової інформації на базі існуючої є **дедукція**, тобто висновок від загального до часткового. Такий дедуктивний підхід є бездоганим з логічного погляду. В його основі лежать транзитивність імплікації (якщо з  $a$  випливає  $b$ , з  $b$  випливає  $c$ , то з  $a$  випливає  $c$ ) і дія квантора загальності (якщо деяка властивість  $P$  виконується для будь-якого елемента множини  $M$ , а  $x \in M$ , то  $P$  виконується для  $x$ ).

Але вербально дедуктивне задання знань не є повним, оскільки:

- дедуктивний висновок не виступає єдино можливим. Мислення людини багато в чому є рефлексивним, інтуїтивним. Воно зазвичай спирається на підсвідомі процеси. Людина часто робить висновки за аналогією, асоціацією. Ці висновки не завжди правильні, але вони істотно доповнюють процеси дедуктивного мислення. Без підсвідомого мислення стає неможливим (будь-яке відкриття зазвичай підсвідоме породження гіпотези, яка потім перевіряється дедуктивним або експериментальним шляхом);
- далеко не всі знання є вербальними. Зокрема, жодне твердження не зберігається в пам'яті людини явно. Відомо, що в основі діяльності мозку людини лежить передача сигналів між нервовими клітинами. Часто людина не може сформулювати свої знання. Наприклад, будь-хто знає, що таке «стіл». Але якщо попросити людину дати визначення цього поняття, у неї можуть виникнути проблеми. Таким чином, поняттями можна оперувати і не знаючи чіткого їхнього визначення. Типовими є слова: «Я не можу пояснити чому, але мені здається».

Тому необхідно розвивати інші моделі знань, окрім вербально-дедуктивних. Наприклад, у межах конекціоністського підходу можна розглядати моделі на основі однорідного поля знань. Однорідним полем знань називається сукупність простих однорідних елементів, які обмінюються між собою інформацією, а нові знання народжуються на основі певних процедур, визначених над полем знань.

У межах вербально-дедуктивної парадигми можна навести таке визначення знань.

**Знаннями** інтелектуальної системи називається трійка  $\langle F, R, P \rangle$ , де  $F$  – сукупність явних фактів, які зберігаються в пам'яті системи в явному вигляді,  $R$  – сукупність правил виведення, які дозволяють на основі відомих знань набувати нових знань,  $P$  – сукупність процедур, які визначають, як потрібно застосовувати правила виведення.

### 3.2 Дані та знання

**Базою знань** (далі – БЗ) інтелектуальної системи називатимемо сукупність усіх знань, що зберігаються в пам'яті системи.

У базах знань прийнято розрізняти екстенціональну та інтенціональну частини.

Екстенціональною частиною бази знань називається сукупність усіх явних фактів, інтенціональною частиною – сукупність усіх правил виведення та процедур, за допомогою яких з існуючих фактів можна виводити нові твердження.

Відомо, що сучасний етап розвитку інформатики характеризується еволюцією моделей даних у напрямі переходу від традиційних (реляційна, ієрархічна, мережева) до моделей знань. Знання, як і будь-яку інформацію, можна вважати даними. Але знання є високоорганізованими даними, для яких характерна певна внутрішня структура та розвинуті зв'язки між різними інформаційними одиницями. Іншим принциповим аспектом є те, що інформаційні системи, які ґрунтуються на знаннях, повинні мати можливість отримувати нові знання на основі існуючих. У цьому можна пересвідчилися на прикладах дедуктивного виведення нових знань з явних фактів.

**Екстенціональна частина БЗ** складається з фактів, які запам'ятовуються явно, інтенціональну – з правил, які дозволяють отримувати нові факти. Наявність розвинутої інтенціональної частини є однією з основних рис, які відрізняють знання від традиційних даних.

**Інтенціональним відношенням**, яке описує базу даних, часто називають правило або сукупність правил, яким підпорядковується кожний запис у базі даних.

Розглянемо, наприклад, базу даних (далі – БД) деканату, що містить дані про те, які курси прослухав кожний студент. Відомо, що жоден студент не має права слухати курс «Бази даних», якщо він не прослухав курсу «Основи програмування». Нехай у базі даних зберігається інформація про Іваненко, Петренко та Біленко, які прослухали обидва курси.

*Екстенціональна частина:*

Прізвище	Дисципліна
Іваненко	Бази даних
Петренко	Бази даних
Біленко	Бази даних

Інтенціональна БД (або її вже можна назвати БЗ) може мати такий вигляд:

Правило: Якщо Прослухав (X, Бази даних), то Прослухав (X, Основи програмування).

Наявність такого правила дозволяє скоротити базу знань: твердження, істинність яких можна встановити за допомогою правил, не обов'язково запам'ятовувати в явному вигляді.

Пошук в інтенціональній БЗ складається з двох етапів:

- пошук потрібного факту в екстенціональній частині;
- дедуктивне виведення факту на основі правил інтенціональної частини.

Реалізовувати інтенціональні правила можна навіть у межах реляційної

моделі даних шляхом програмування відповідних запитів.

Знання можуть бути **неповними**. Це означає, що для доведення або спростування певного твердження може не вистачати інформації. У багатьох системах логічного виведення прийнято **постулат замкненості світу**: на запит про істинність деякого твердження система відповідає «так» тоді і тільки тоді, коли його можна довести; якщо ж довести неможливо, система відповідає «ні». Водночас *«неможливо довести через нестачу інформації»* і *«доведено, що ні»* – це зовсім не те саме. З огляду на це бажано, щоб експертна система запитувала у користувача про факти, яких не вистачає.

Знання можуть бути **недостовірними**. Наприклад, на виготовлення продукції можуть впливати випадкові чинники (об'єктивна невизначеність) або експерт може бути не зовсім упевненим у деякому факті чи правилі (суб'єктивна невизначеність). Ненадійність знань і недостовірність наявних фактів обов'язково повинні враховуватися в процесі логічних побудов. Звичайно, можна було б просто відкидати факти та правила виведення, які викликають сумнів, але довелося би відмовитися від цінної інформації. Крім того, в експертних системах часто доводиться мати справу з неточно визначеними поняттями, такими, як *«великий»*, *«маленький»* тощо.

### **Зв'язки між інформаційними одиницями**

У базі знань можуть зберігатися відомості про різні об'єкти, поняття, процеси тощо, тобто відповідні описи, які називаються **інформаційними одиницями**.

*Опис, який утворює інформаційну одиницю, розглядається як деяка абстракція реальної сутності, що існує в дійсному світі.*

Це означає що ця абстракція описує певні риси реальної сутності. Бажано, щоб абстракція описувала найважливіші з них.

У реальному світі все тісно взаємопов'язане. Відповідні зв'язки повинні знайти адекватне відображення в БЗ. Існує велика група зв'язків, за допомогою яких можна описувати просторові відношення («знизу», «зверху», «поруч», «між» і т. п.); часові відношення («раніше», «пізніше», «під час», «одночасно» і т. п.), причинно-наслідкові відношення тощо. Ці відношення є спільними для всіх предметних областей.

Логічні системи, що ґрунтуються на цих відношеннях, називаються **псевдофізичними логіками**. Використання псевдофізичних логік (так само, як і більш загальних предикатів, таких, як узагальнення та агрегація) дає можливість інтелектуальній системі поповнювати первинні описи.

Основними зв'язками між інформаційними одиницями є **узагальнення** та **агрегація**.

Агрегація дозволяє задавати будову об'єкта та його складових. Наприклад, агрегація дає змогу визначити в базі знань аудиторію як об'єкт, що має вікна, двері і т. п. Вікна, зі свого боку, теж можна розглядати як агрегований об'єкт: вони мають ручки, рами і т. п. Інакше кажучи, відношення *«Має»*, яке описує агрегацію, означає, що певний об'єкт містить у своєму складі деякий інший об'єкт. За цим відношенням закріпилася спеціальна назва *«Has\_Part»*. По суті, це

відношення «ціле – частина».

В усіх системах керування базами знань має бути забезпечена належна підтримка успадкування (інша назва – виведення **за наслідуванням**). Суть механізму успадкування можна сформулювати так: якщо деяка умова виконується для всього класу, то вона виконується і для кожного представника цього класу, а також для всіх підкласів цього класу (якщо інше не задано у явний спосіб). Інакше кажучи, екземпляри успадковують властивості класів, підкласи успадковують властивості надкласів.

### 3.3 Властивості та моделі знань

Сформулюємо особливості знань, які відрізняють їх від звичайних даних:

1) **внутрішня інтерпретованість**: кожна інформаційна одиниця повинна мати унікальне ім'я, за яким інформаційна система її знаходить, а також відповідає на запити, у яких це ім'я згадується;

2) **структурованість**: знання повинні мати гнучку структуру; одні інформаційні одиниці можуть включатися до складу інших (відношення типу «частина – ціле», «елемент – клас»);

3) **зв'язність**: в інформаційній системі повинна бути передбачена можливість встановлення різних типів зв'язків між різними інформаційними одиницями (причинно-наслідкові, просторові та ін.);

4) **семантична метрика**: на множині інформаційних одиниць корисно задавати відношення, які характеризують ситуаційну близькість цих одиниць;

5) **активність**: виконання програм в інтелектуальній системі повинно ініціюватися поточним станом бази знань.

Часто виокремлюють інші властивості знань, наприклад **шкальованість**, яка означає, що формально неоднакові поняття насправді відображаються на одній і тій самій **шкалі понять**, різні точки якої відповідають інтенсивності того самого фактора. Наприклад, температура може бути високою або низькою, і це породжує такі поняття, як «холодно», «тепло», «гаряче» тощо.

*Моделлю знань називається фіксована система формалізмів (понять і правил), відповідно до яких інтелектуальна система подає знання в своїй пам'яті та здійснює операції над ними.*

Моделі задання знань необхідні:

- для створення спеціальних мов описів знань і маніпулювання ними;
- для формалізації процедур зіставлення нових знань з уже існуючими;
- для формалізації механізмів логічного виведення.

Відомими моделями, які ґрунтуються на вербально-дедуктивній парадигмі, є чотири класи моделей:

- семантичні мережі;
- фреймові;
- логічні;
- продукційні.

Вербально-дедуктивні моделі задання знань мають багато спільних рис, Отже, можна вважати, що всі вони мають єдину концептуальну основу.

### 3.4 Неоднорідність знань. Галузі та рівні знань

Для успішного здійснення операцій зі знаннями необхідно відокремлювати в БЗ певні фрагменти, які називаються **областями знань**. Ці області знань повинні бути відносно незалежними між собою, що означає таке:

- зміни в одній області знань не повинні приводити до суттєвих змін у інших областях;
- вирішення складної задачі можна зазвичай звести до підзадач так, що для вирішення кожної з цих підзадач достатньо знань з однієї області.

Такий розподіл є важливим для полегшення проектування і використання БЗ. Зокрема, різні області знань можуть проектуватися незалежно одна від одної.

Виходячи з постановки задачі можна по-різному розділяти знання на області. Зокрема, для експертних систем, які ведуть діалог з користувачем мовою, наближеною до природної, можна виокремити такі області знань [3–4]:

- **предметна область**, яка містить знання про конкретну предметну галузь, в якій працює експертна система;
- **область мови**, яка містить знання про мову, якою ведеться діалог;
- **область системи**, яка містить знання експертної системи про власні можливості;
- **область користувача**, яка містить знання про користувача. Наявність їх дозволяє враховувати індивідуальні особливості кожного користувача. Наприклад, пояснення користувачеві можуть надавати залежно від рівня його підготовленості;
- **область діалогу**, яка містить знання про мету діалогу, а також про форми та методи його організації. Знання можуть різнитися за **рівнем задання і рівнем детальності**.

За **рівнями задання** розрізняють знання нульового рівня (конкретні і абстрактні) і знання вищих рівнів – знання про знання (**метазнання**).

У випадку класифікації знань за **рівнями детальності** до уваги береться ступінь деталізації знань. Кількість рівнів детальності залежить від специфіки задачі, обсягу знань і моделі їхнього задання. Якщо користувач запитує в експертної системи, як здійснити певну операцію. У разі отримання відповіді на найвищому рівні детальності система видає *загальний план*. Якщо цей загальний план не є достатнім, система може спуститися на нижчий рівень і розписати операції детальніше.

### 3.5 База знань як об'єднання простіших одиниць

БЗ є кон'юнкцією простих тверджень. Можна вважати, що кожне з них відповідає окремому реченню природної мови. Подібні твердження називають концептуальними одиницями. Останні можуть бути або фактами, або правилами виведення.

Якщо концептуальна одиниця становить собою факт, вона може бути описана певним **предикатом**, тобто логічною функцією, що залежить від заданої кількості змінних і може приймати одне з двох можливих значень 0 або 1 (false або true). Конкретні твердження утворюються з предикатів шляхом підстановки конкретних значень аргументів. При цьому для опису одного й того самого

твердження можна використовуватися різні предикати.

### **Бінарні предикати і тріада «об'єкт – атрибут – значення».**

Концептуальна одиниця може бути достатньо складною. Тому концептуальну одиницю, що є фактом, часто зручно розглядати як кон'юнкцію елементарніших тверджень, а саме – **бінарних фактів**, кожний з яких описується **бінарним предикатом**, тобто предикатом, який залежить від двох змінних. Якщо формалізувати правила об'єднання бінарних предикатів у складніші одиниці, то на основі бінарних предикатів можуть бути сконструйовані складні твердження та системи знань.

### **Структура бази знань**

Структурно БЗ можна організувати у вигляді двох основних підбаз – бази правил (БП) і бази даних (БД).

У БД зберігається фактографічна інформація про розв'язувані на об'єкті завдання і дані, які відносяться до зазначеної предметної області. БП визначає відносини між елементами даних, що зберігаються в БД, на основі моделей подання знань про предметну область, а також способи активізації цих знань.

Таким чином, дуже узагальнено можна говорити про два рівні подання знань: перший рівень – фактографічна інформація, дані; другий рівень – опису, відносини, правила й процедури, що визначають спосіб маніпулювання фактографічної інформації.

Крім знань про предметну область, у БЗ повинні зберігатися й інші типи знань: модель миру системи, знання про користувача, цілі тощо. Ці знання переважно втримуються на другому рівні подання у вигляді блоків або органічних частин БП.

У багатьох інтелектуальних системах [4] реалізується рівень метазнань, що необхідний для забезпечення раціоналізації процесів оперування знаннями в БЗ, зменшення області пошуку рішення, скорочення часу обробки інформації. Метазнання – це знання системи про себе, тобто знання про свої знання, їхній структурі й про принципи свого функціонування. На основі цих знань на рівні метазнань (у блоці метазнань) серед наявного набору стратегій пошуку визначається найефективніша (рис. 3.1).

На рисунку 3.2. наведено фрагмент системи П, що представляє взаємодію БЗ із основними модулями системи при пошуку й генерації знань [4]. У БЗ подана як фактографічна інформація, так і правила, або евристики.

Вивід рішення або генерація нових правил і знань здійснюється за допомогою блока виводу, що взаємодіє з метарівнем БЗ при інтерпретації правил і даних БЗ.

Рішення завдання й робота із правилами й даними здійснюються в спеціальному блоці – робочій області. У робочій області представляються описи запиту – або розв'язуваного завдання, дані й правила із БЗ, процедури або стратегія механізму виводу.

За використання найпоширеніших у цей час продукційних систем подання знань можливий варіант структурної й функціональної організації основних компонентів системи, поданий на рисунку 3.3.



Рисунок 3.1 – Узагальнена структура БЗ

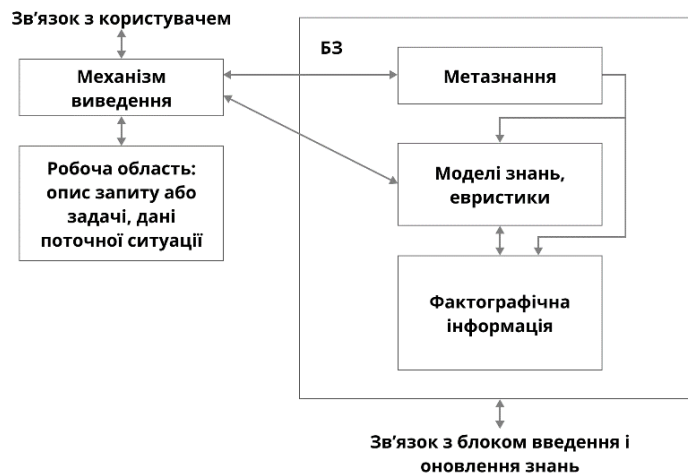


Рисунок 3.2 – Структура взаємодії БЗ із модулями системи при пошуку й генерації знань

### 3.6 Моделі подання знань в інтелектуальних інформаційних системах

Важливим питанням під час створення БЗ є вибір способу подання знань. Мета подання знань – організація необхідної інформації в таку форму, щоб програма штучного інтелекту мала легкий доступ до неї для ухвалення рішень, планування, дізнання об'єктів і ситуацій, аналізу сцен, виводу висновків і інших когнітивних функцій.

Подання знань в інтелектуальних системах здійснюється на основі:

1. Фреймів і семантичних мереж.
2. Продукційних і логічних моделей.
3. Моделей подання й формалізації нечітких знань.
4. Нейронних мереж.

На рисунку 3.3 представлена структура взаємодії БЗ із основними компонентами ІС для продукційних систем [4].



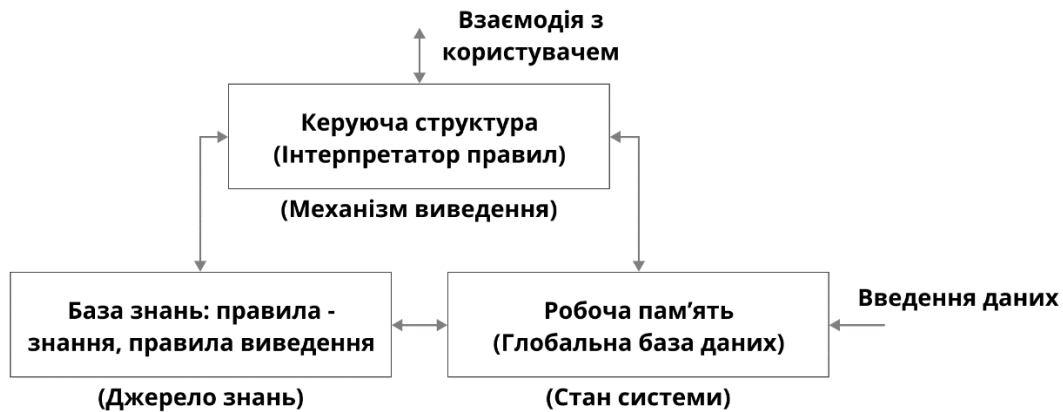


Рисунок 3.3 – Структура взаємодії БЗ із основними компонентами ІС для продукційних систем

Знання в ІС можна подати за допомогою моделей двох типів: декларативних і процедурних. До типових декларативних моделей відносять семантичні мережі й фрейми, а типовим процедурним моделям – вираження предикатів, системи продукцій, нечітка логіка. На практиці рідко вдається обійтися рамками однієї моделі при розробці ІС, тому подання знань виходить складним.

*Семантична мережа* становить орієнтований граф, вершинами якого є інформаційні одиниці, що мають індивідуальні імена. Як інформаційна одиниця можуть виступати події, дії, узагальнені поняття або властивості об'єктів. Вершини графа з'єднуються дугою, якщо відповідні інформаційні одиниці перебувають у якому-небудь відношенні.

Фрейм становить структуру даних, що дає цілісне подання про об'єкти, явища і їхні типи у вигляді абстрактних образів. Структура фрейму записується у вигляді списку властивостей (слотів). Кожний фрейм має спеціальний слот, який заповнений найменуванням, і представляється суттю, а інші заповнені значеннями різноманітних атрибутів, що асоціюються з об'єктом.

Логіка предикатів є розширенням логіки висловлень. Основним об'єктом тут є змінне висловлення (предикат), істинність і хибність якого залежать від значення його змінних. Мова логіки предикатів є потужнішою порівняно з мовою логіки висловлень. Вона придатна для формалізації понять багатьох проблемних областей.

Продукційна модель, або модель, заснована на правилах, дозволяє подати знання у вигляді пропозицій типу ЯКЩО (умова), ТО (дія).

### **Обробка знань і вивід рішень в ІС**

Основними методами обробки знань і виводу рішень в ІС є:

I. Методи виводу й пошуку рішень у продукційних системах.

1. Методи виводу на основі прямого й зворотного ланцюжків.
2. Загальні методи пошуку рішень у просторі станів.
  - o Методи перебору.

- Пошук у глибину.
  - Пошук завширшки.
  - Пошук на основі вартості дуг (Знаходження шляхів мінімальної вартості).
  - Пошук з поверненням (бектрекинг).
3. Евристичні методи пошуку (для визначення напрямку пошуку використовується оцінна функція).
  4. Методи редукції.
  5. Методи пошуку рішень у більших просторах станів.
    - Методи породження й перевірки.
    - Методи послідовного уточнення зверху.
- II. Виводи на фреймах і в семантичних мережах.
- III. Дедуктивні методи пошуку рішень.
- IV. Пошук рішень в умовах невизначеності.
1. Імовірнісний вивід.
  2. Вивід на основі теорії впевненості.
  3. Нечітка логіка й наближені міркування.
- V. Вивід у нейронних мережах.

### **Контрольні запитання**

1. Дайте визначення поняттю «знання».
2. Що становить база знань?
3. Сформулюйте поняття «експертні системи».
4. Яка різниця між даними і знаннями?
5. Що таке інформаційна одиниця?
6. Дайте визначення «моделі знань».
7. Опишіть структуру бази даних.
8. Назвіть моделі подання знань в ПС.
9. Які методи використовуються під час обробки знань?
10. Опишіть проєкційну модель подання знання.

## **ТЕМА 4 КЕРУВАННЯ ІНТЕЛЕКТУАЛЬНИМИ СИСТЕМАМИ**

### **4.1 Алгоритмічний та декларативний підходи до керування**

На найбільш загальному рівні, можна виділити два підходи до керування складними системами та до програмування роботів і комп'ютерів, що могли б розв'язувати ті чи інші задачі. Під час програмування сучасних комп'ютерів переважно реалізований традиційний **алгоритмічний підхід**, який можна ще назвати імперативним. Цей підхід вимагає заздалегідь продумати та детально розписати, як потрібно вирішувати певну проблему. Написання програми вимагає задання чітких послідовностей інструкцій. Якщо таку послідовність вдається написати – комп'ютер зможе вирішувати цю задачу, якою б складною

вона не була. Але, зрозуміло, він виконує інструкції, абсолютно не розуміючи їх змісту. Якщо зміняться умови, за яких виконується програма, комп'ютер може виявитися безпомічним [4].

Інший підхід – **декларативний**. Інтелектуальному виконавцеві (людині чи комп'ютерові) достатньо сказати, **що** потрібно робити, тобто лише сформулювати завдання, побудувавши всі взаємозв'язки між об'єктами у предметній області. **Як** це завдання буде виконуватися – повинен визначити сам виконавець.

**Приклади.** 1. Запустити космічний корабель так, щоб він приземлився на Марсі, взяв зразки ґрунту та привіз їх назад – задача дуже складна, але вона піддається точній алгоритмізації. Математичні методи дозволяють точно розрахувати траєкторію, по якій повинен рухатися такий корабель.

2. Послати робота до магазину за пляшкою молока – задача на декілька порядків більш складна. Не кажучи вже про сам процес спілкування з продавцем, робот повинен вирішити низку не зовсім формалізованих підзадач. Заздалегідь розрахувати траєкторію руху неможливо, оскільки робот повинен уникнути зіткнень з людьми та автомобілями. Якщо магазин закритий на переоблік, він повинен знайти інший магазин. *Неможливо передбачити всі ситуації, які можуть виникнути, а відтак – алгоритмізувати розв'язок задачі.* Тому виконання такого завдання під силу лише інтелектуальній системі, яка вміє *орієнтуватися в зовнішньому світі, аналізувати поточні ситуації та коригувати, адаптувати свою поведінку на основі такого аналізу.*

#### **4.2 Формалізація понять алгоритмічності та декларативності**

Спробуємо формалізувати деякі з впроваджених раніше понять. Введемо такі позначення:

$q$  – *первинні інструкції*, записані без будь-яких змін у тому вигляді, у якому їх сформулював автор; аналогічно можна визначити *первинний опис ситуації* як результат її безпосереднього сприйняття;

$S$  – множина факторів, які визначають поточний стан виконавців; розглядатимемо  $S$  як об'єднання двох множин:  $S_1$  та  $S_2$ , де  $S_1$  – множина *контрольованих факторів*, які відомі авторові процедури і на які він може мати вплив,  $S_2$  – множина *неконтрольованих факторів*;  $Z$  – знання, які має виконавець;

$r_A$  – робочий алгоритм, який формується та реалізується виконавцем при алгоритмічному підході. Разом із тим зазвичай автоматично формується і програма  $p_A$ , що відповідає цьому алгоритму;

$r_D$  – робочий алгоритм, який формується та реалізується інтелектуальним виконавцем при декларативному підході. Згідно з фундаментальними тезами Тьюринга та Черча (все, що може бути виконано будь-яким виконавцем, може бути промодельоване на машині Тьюринга) сам факт виконання завдання свідчить про існування цього алгоритму [4, 5]. Щоправда, цей алгоритм може і не усвідомлюватися виконавцем, а тому не може бути явно сформульований у вигляді алгоритму чи програми (якщо ж програма, що відповідає алгоритму  $r_D$ , повинна бути згенерована, йдеться мова про *задачу синтезу програм*).

Тоді можна записати такі співвідношення:

$$r_A = f(q, S_1),$$

$$r_D = g(q, S_1, S_2, z).$$

Принциповим є таке твердження. Алгоритм  $r_A$  формується на основі первинних інструкцій  $q$  однозначно. При цьому також можуть відбуватися зміна і поповнення первинних інструкцій, але цей процес має повністю контрольований і детермінований характер. Як приклад можна навести компіляцію програм, написаних мовами високого рівня. Тому автор процедури може бути впевнений у гарантованому результаті (якщо, звичайно, були дотримані певні формальні вимоги до первинних інструкцій і забезпечено належний стан виконавця).

На противагу цьому, за декларативного підходу такої впевненості **немає**. Інтелектуальний виконавець у певний спосіб поповнює первинні інструкції, але їх авторові не завжди відомо, як це поповнення відбувається. Тому не можна бути впевненим у результаті виконання інструкцій, і ця втрата гарантованості є неминучою платою за відмову від алгоритмічності. Отже, ми маємо справу з узагальненням поняття алгоритму, яке дістало назву «**квазіалгоритм**».

Алгоритмом називається чітка однозначна зрозуміла виконавцеві послідовність інструкцій, виконання яких обов'язково приводить до гарантованого результату за скінченний час. На відміну від цього, інструкції квазіалгоритму можуть бути не зовсім чіткими, і результат виконання квазіалгоритмічної процедури не обов'язково гарантованим.

Можна виділити як мінімум чотири основні *джерела квазіалгоритмічності*:

1. **Дія випадкових чинників**, що не залежать від виконавця. Строго кажучи, з огляду на цей фактор квазіалгоритмом варто вважати будь-який, навіть найбільш формалізований, алгоритм. Алгоритм розрахований на роботу за певних умов, якщо ці умови зміняться, алгоритм може не спричинити потрібного результату. *Якщо ж не вдається чітко окреслити межі застосування алгоритму або забезпечити виконання необхідних умов для його роботи, ми маємо справу зі справжнім квазіалгоритмом.*

2. **Недостатнє врахування автором алгоритмічної процедури особливостей виконавця**. Це призводить до того, що виконавець неправильно розуміє, що від нього вимагається. Процедура може бути в принципі алгоритмічною, за нормальних умов спричиняти гарантований результат, але цей результат може бути не передбачений автором і тому бути для нього несподіванкою.

3. **Нечіткість формулювань**, що фігурують в описі процедури. Приклад. Розглянемо будь-який кулінарний рецепт, наприклад: «розтерти тісто, змішати з дрібно нарізаними яблуками, додати солі і перцю за смаком і смажити до появи рум'яної скоринки». Це є типовим прикладом нечіткості формулювань, що призводить до невизначеностей. До якої межі потрібно розтирати тісто? Що означає «дрібно нарізані»? Що означає «за смаком» тощо.

Неінтелектуальна система, орієнтована на суто алгоритмічне керування, просто не зрозуміє цього опису і тому не зможе його виконати. Інтелектуальна ж система, наприклад, людина, повинна спробувати поповнити і уточнити цей опис на підставі наявних знань і досвіду. Сам по собі факт виконання завдання буде свідчити про те, що квазіалгоритмічний первинний опис процедури був зведений до деякого внутрішнього алгоритму, але навряд чи виконавець зможе чітко пояснити і розписати цей алгоритм. Крім того, ці внутрішні алгоритми для кожного виконавця будуть різними.

«Свобода волі» можуть мати високоорганізовані, по-справжньому інтелектуальні системи. Ці системи можуть мати свої цілі, і, якщо ця процедура суперечить цим цілям, вони можуть відмовитися її виконувати або виконати не так, як це від них вимагається. «Свобода волі» полягає у тому, що інтелектуальна система самостійно ухвалює рішення про своє поведіння і залежно від цих рішень може виконувати зовнішні розпорядження, не виконувати їх, або ж виконувати так, як вона вважає за потрібне.

#### **4.3 Характеристика інтелектуальних систем з погляду кібернетики**

Інтелектуальна система може припускати зовнішнє керування, але для неї характерною є **самокерованість**. Система має певну мету і прагне так **планувати свої дії, щоб досягати цієї мети**. Як вхідні стимули системи можна розглядати поточну ситуацію, що сприймається і аналізується системою. Результатом реакції системи стає зміна зовнішньої ситуації, і поведіння системи коригується залежно від того, бажаною чи небажаною є ця зміна.

Людина має певну суму **знань про світ**, яка дозволяє їй орієнтуватися в життєвих ситуаціях та ухвалювати правильні рішення. Крім того, людина вмє певним чином **використовувати ці знання**. Ці самі риси мають мати системи штучного інтелекту.

Також можна стверджувати, що **здатність до поповнення первинних знань є однією із ключових рис інтелектуальних систем**. Ця властивість інтелектуальних систем називається **здатністю до навчання**. Розрізняють *зовнішнє навчання та самонавчання*.

Підсумувавши усі сказане, можна стверджувати, що **інтелектуальною системою називається самокерована кібернетична система, яка має певну суму знань про світ і здатна на основі безпосереднього сприйняття і подальшого аналізу поточної ситуації до планування дій, спрямованих на досягнення мети, а також до навчання**.

**Типова схема функціонування** інтелектуальної системи.

Функціонування інтелектуальної системи можна описати як **постійне** ухвалення рішень на основі аналізу поточних ситуацій для досягнення певної мети. Схема функціонування інтелектуальної системи наведена на рисунку 4.1.

Виокремлюють такі етапи функціонування інтелектуальних систем.

1. **Зіставлення первинного опису зі знаннями системи і поповнення цього опису**. Результатом є формування вторинного опису ситуації в термінах знань системи. Цей процес можна розглядати як процес *розуміння ситуації* або як процес перекладу первинного опису на внутрішню мову системи. При

цьому можуть змінюватися внутрішній стан системи та її знання.

Вторинний опис може бути **не єдиним**, і система має змогу вибирати між різними вторинними описами. Крім того, система в процесі роботи **може** переходити від одного вторинного опису до іншого. Якщо ми можемо формально задати форми внутрішнього представлення описів ситуацій та операції над ними, то сподіватимемося на певний автоматизований аналіз цих описів.

Результатом є первинний опис ситуації.

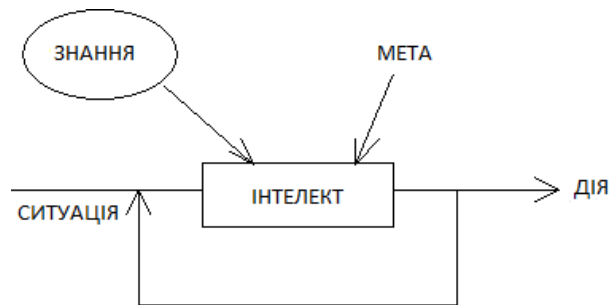


Рисунок 4.1 – Схема функціонування інтелектуальної системи

**2. Зіставлення первинного опису зі знаннями системи і поповнення цього опису.** Результатом є формування вторинного опису ситуації в термінах знань системи. Цей процес можна розглядати як процес *розуміння ситуації* або як процес перекладу первинного опису на внутрішню мову системи. При цьому можуть змінюватися внутрішній стан системи та її знання.

Вторинний опис може бути не єдиним, і система має змогу вибирати між різними вторинними описами. Крім того, система в процесі роботи може переходити від одного вторинного опису до іншого. Якщо ми можемо формально задати форми внутрішнього представлення описів ситуацій та операції над ними, то сподіватимемося на певний автоматизований аналіз цих описів.

**3. Планування цілеспрямованих дій та ухвалення рішень, аналіз можливих дій та їхніх наслідків і вибір тих дій, що найкраще узгоджуються з метою системи.** Таке рішення зазвичай формулюється певною внутрішньою мовою (свідомо або підсвідомо).

Зворотна інтерпретація ухваленого рішення, тобто формування робочого алгоритму для реагування системи.

**4. Реалізація реакції системи; наслідками є зміни зовнішньої ситуації і внутрішнього стану системи тощо.**

Проте не варто вважати, що зазначені етапи є повністю відокремленими у тому розумінні, що наступний етап починається тільки після того, як повністю закінчиться попередній. Навпаки, для функціонування інтелектуальної системи характерним є взаємне проникнення цих етапів. Наприклад, ті чи інші рішення можуть ухвалюватися вже на етапі безпосереднього сприйняття ситуації.

## Контрольні запитання

1. Сформулюйте поняття алгоритмічності та декларативності.
2. Що таке «квазіалгоритм»?
3. Надайте характеристику інтелектуальних систем з погляду кібернетики.
4. Опишіть схему функціонування інтелектуальної системи.

## ТЕМА 5 МЕТОДИ І СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ

### 5.1 Основні положення теорії штучного інтелекту

Інтелект – здатність людини сприймати інформацію від зовнішнього середовища, оцінити ситуацію та ухвалювати адекватні рішення.

Системою штучного інтелекту називається **людино-машинна система**, яка використовується для **автоматизації процесів** творчої діяльності людини в різних сферах, зокрема ухваленні рішень [1–5].

Термін «штучний інтелект» з'явився в 1956 році (семінар у Дортмунді).

Теорія штучного інтелекту як наука розділена на **дві** складові: **нейрокібернетика** та **кібернетика «чорної скрині»**.

Нейрокібернетика започаткована роботами Мак-Калока (1956 р.) та Розенблата (1962 р.), за якими була запропонована модель **перцептрона**. Особливий розвиток нейрокібернетики спостерігався в 1980-х роках.

Напрямок «чорної скриньки» будується на принципі, що **структура інтелектуальної системи неважлива**, головне щоб ІС реагувала на ситуації так само, як людина. Цей напрямок був орієнтований на автоматизований пошук алгоритмів вирішення інтелектуальних задач. Завдяки цього напрямку з'явилися мови штучного інтелекту (МакКарті, ЛІСП) та фреймові моделі (Мінський).

### Основні класи задач штучного інтелекту

До основних класів задач штучного інтелекту відносять:

- подання знань, маніпуляції знаннями та створення експертних систем;
- спілкування, комунікації «людина – комп'ютер»;
- розпізнавання образів;
- навчання та самонавчання;
- планування дій, пошук розв'язків задач;
- нейронні мережі;
- самоорганізація, методи евристичної самоорганізації;
- еволюційне моделювання та генетичний алгоритм;
- автоматизація конструювання та проектування.

### Математичні моделі задач штучного інтелекту

У загальному математичному розумінні, інтелектуальна задача задається у формі:

$P = \langle S, C, D, M, L, W \rangle$ ;

S – множина можливих ситуацій;

C – множина керувань для кожної ситуації;

- D – множина завад;
- M – множина переходів від однієї ситуації у іншу;
- L – множина бажаних ситуацій;
- W – множина небажаних ситуацій.

## 5.2 Системи розпізнавання образів

Кількісні дані (знання) можуть бути неточними. Для обліку неточності лінгвістичних знань використовується формальний апарат нечіткої алгебри. Одне з головних понять у нечіткій логіці – це поняття лінгвістичної змінної, котре визначається через нечіткі множини. Нечіткі множини дозволяють урахувати суб'єктивні думки окремих експертів.

Нейронні мережі – це напрямок комп'ютерної індустрії, в основі якого лежить ідея створення її за образом й подобою людського мозку. Існує велика кількість різних алгоритмів навчання нейромереж, серед яких успішним вважається ідея **генетичних алгоритмів**, що складається в імітації природних оптимізаційних процесів, що відбуваються при еволюції живих організмів.

Конекціоністський підхід [4] до побудови систем штучного інтелекту розвинувся на противагу символічному, що є характерним для сучасних моделей знань. В основі конекціоністського підходу лежить спроба безпосереднього моделювання розумової діяльності людського мозку. Відомо, що мозок людини складається з величезної кількості нервових клітин (нейронів), що взаємодіють між собою. Ці «обчислювальні елементи» мозку **функціонують набагато повільніше**, ніж обчислювальні елементи комп'ютерних систем. Але ефективність людського інтелекту досягається як шляхом паралельної роботи нейронів, так і завдяки тому, що механізми їхньої взаємодії були вироблені шляхом тривалої еволюції.

Для багатьох цілей нейрон можна розглядати як елемент з певним критичним значенням. Це означає, що він або ж дає на виході деяку постійну величину, якщо сума його входів досягає певного значення, або ж залишається пасивним.

Мак-Каллок і Пітс довели, що будь-яку обчислювану функцію можна реалізувати за допомогою спеціально організованої мережі **ідеальних** нейронів, логічні властивості яких з високою достовірністю можна приписати реальному нейрону. Але ця мережа буде мати такі вади.

По-перше, проблема полягає в тому, чи можна знайти якийсь розумний принцип реорганізації мережі, який дозволяв би випадково об'єднувати спочатку групи ідеальних нейронів, які потім самоорганізуються в «обчислювальний пристрій», здатний розв'язувати довільну задачу розпізнання.

По-друге, потрібно використовувати велику кількість нейронів. Наприклад, модель мурашки потребує використання близько 20 000 нейронів, людини – 100 млрд нейронів, що на практиці неможливо.

Нейрологічна теорія стала також основою системи розпізнавання, яка дістала назву «**перцептрон**». У цьому підході основна увага приділялась встановленню характеристик, приписаних фіксованій множині детекторів ознак. Альтернативний підхід розпізнавання зводиться до пошуку «**добрих**» ознак, на



основі яких розпізнавання здійснюється найбільш чітко. Наприклад, перцептрон Розенблатта передавав повідомлення від «ока», яке реалізовувалось системою фотоелементів, в блоки електромеханічних комірок пам'яті, які оцінювали відносні величини електричних сигналів. Ці комірки з'єднувались між собою випадковим чином, створюючи мережу з прямими зв'язками. У ній були відсутні зворотні зв'язки між нейроподібними елементами. Перцептрон міг навчатись шляхом спроб і помилок, а також корекцією електричних імпульсів.

Мінські і Пейпертом було математично доведено, що перцептрони **не** в змозі виконувати багато приписуваних їм функцій, наприклад, розпізнавання частково затулених предметів. Після цього результату розвиток перцептронної теорії призупинився.

### **5.3 Нейрокомп'ютер – програмно-технічна система**

Один з сучасних напрямків створення розумних машин – це розробка *нейрокомп'ютерів*. *Нейрокомп'ютер* – це програмно-технічна система (спеціалізована ЕОМ), яка реалізує деяку формальну модель природної мережі нейронів.

В основу машин п'ятого покоління покладено ідею паралельної обробки інформації в нейроподібних системах. Незважаючи на те, що електронний процесор працює в тисячі разів швидше, ніж його нейронний еквівалент у мозку, мережі нейронів розв'язують багато задач (особливо нечислових) в тисячі раз швидше, ніж електронний процесор.

Причини цього такі:

1. Характер взаємозв'язків між нейронами дозволяє розв'язувати багато задач на основі паралельної обробки.
2. У нейронній мережі пам'ять не локалізована в одному місці (як в послідовних машинах), а розподілена по всій структурі. У біологічних системах пам'ять реалізується підсиленням або послабленням зв'язків між нейронами, а не зберіганням двійкових символів.
3. Біологічні мережі реагують не на всі, а тільки на визначені зовнішні подразнення. Кожний нейрон виступає як елемент ухвалення рішення і як елемент зберігання інформації. Перевага такої структури – «життєздатність» (вихід з ладу декількох нейронів не призводить до значної зміни даних, що зберігаються, або ж до руйнування всієї системи).
4. Можливість адресації за вмістом (асоціативної пам'яті) є ще однією важливою характеристикою систем з розподіленою пам'яттю (кожний елемент відшукується за його вмістом, а не зберігається в комірниці пам'яті з визначеним номером).

В основу зв'язків у нейрокомп'ютерах покладено *принцип асоціацій*. *Асоціативні зв'язки* пронизують все мислення людини. Існує думка [1], що *процеси мислення є не що інше, як розповсюдження певного збудження, як деяка ланцюгова реакція*. Навіть найбільш примітивні процеси навчання принципово залежать від послідовності подій в часі. Це й закладено в природу нейронних систем. Тому їм притаманне реагування тільки на жорстко визначені зовнішні подразнення. Наприклад, домашні тварини «навчаються» ігнорувати повторні

несуттєві зовнішні подразнення («цокання» годинника), але посилюють сприйняття подразнень, які можуть мати серйозні наслідки (звук автомобільних гальм).

Багато дослідників вважає, що майбутнє належить комп'ютерам, які базуються на аналізі зв'язків, а не обробці символів. Мінські вважав, що якщо комп'ютер повинен діяти подібно мозку, тоді й його конструкція повинна бути також подібна до мозку.

Моделі штучних нейронних мереж (ШНМ) і схеми з адресацією за вмістом мають і недоліки. Внаслідок нефіксованої організації вони можуть плутати різні об'єкти. Але це аналогічно звиканню, посиленню чуттєвості до асоціацій, які лежать в основі психічних особливостей людини. Іншими словами, будь-який комп'ютер, який претендує на «розумність», повинен мати такі особливості.

Штучні нейрони, що також називаються нейронними клітинами, вузлами, модулями, моделюють структуру й функції біологічних нейронів. Архітектура й особливості штучних нейронних мереж, утворених нейронами, залежать від конкретних завдань, які мають бути вирішені з їхньою допомогою.

#### 5.4 Модель штучного нейрона

Структуру штучного нейрона [6, 8] зображено на рисунку 5.1.

Вхідними сигналами штучного нейрона  $x_i$  ( $i = 1, N$ ) є вихідні сигнали інших нейронів, кожний з яких узятий зі своєю вагою  $w_i$  ( $i = 1, N$ ), аналогічною досинаптичній силі.

Вхідний оператор  $f_{ВХ}$  перетворює зважені входи й подає їх на оператор активації  $f_a$ . Вихідний сигнал нейрона  $y$  становить перетворений вихідним оператором  $f_{ВІХ}$  вихідний сигнал оператора активації. Таким чином, нелінійний оператор перетворення вектора вхідних сигналів  $x$  у вихідний сигнал  $y$  може бути записаний у такий спосіб

$$y = f_{ВІХ}(f_a(f_{ВХ}(x, w))). \quad (5.1)$$



Рисунок 5.1 – Структура штучного нейрона [8]

Вихідний сигнал цього нейрона є вхідним для наступного.

**Вхідний оператор** (вхідна функція) нейрона задає вигляд використовуваного в нейроні перетворення зважених входів. Відмінність гальмуючих входів від збуджувальних відбивається у знаках відповідних ваг.

Зазвичай використовуються такі вхідні функції:

– сума зважених входів

$$f(\mathbf{x}, \mathbf{w}) = \sum_{i=1} w_i x_i;$$

– максимальне значення зважених входів

$$f(\mathbf{x}, \mathbf{w}) = \max_i (w_i x_i);$$

– мінімальне значення зважених входів

$$f(\mathbf{x}, \mathbf{w}) = \min_i (w_i x_i).$$

**Функція активації**  $f_a(\cdot)$  описує правило переходу нейрона, що перебуває в момент часу  $k$  у стані  $z(k)$ , у новий стан  $z(k+1)$  при надходженні вхідних сигналів  $x$

$$z(k+1) = f_a(z(k), f_{\text{вх}}(\mathbf{x}, \mathbf{w})).$$

Надалі позначатимемо функцію активації без індексу « $a$ ».

На рисунках 5.2 – 5.8 подаються основні функції активації [8].

Найпростішими активаційними функціями є

– лінійна

$$f(z) = Kz, K = \text{const};$$

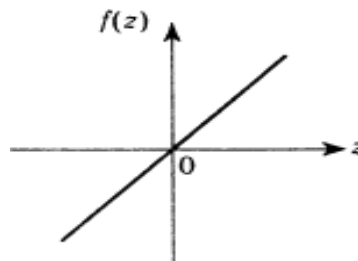


Рисунок 5.2 – Лінійна функція

– лінійна біполярна з насиченням

$$f(z) = \begin{cases} 1 & \text{при } z \geq \alpha_2; \\ Kz & \text{при } -\alpha_1 \leq z \leq \alpha_2; \\ -1 & \text{при } z \leq -\alpha_1; \end{cases}$$

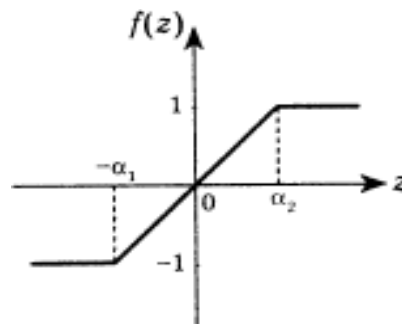


Рисунок 5.3 – Лінійна біполярна функція з насиченням

– лінійна уніполярна з насиченням

$$f(z) = \begin{cases} 1 & \text{при } z \geq \frac{1}{2a}; \\ \alpha z + 0,5 & \text{при } |z| < \frac{1}{2a}; \\ 0 & \text{при } z \leq -\frac{1}{2a}. \end{cases}$$

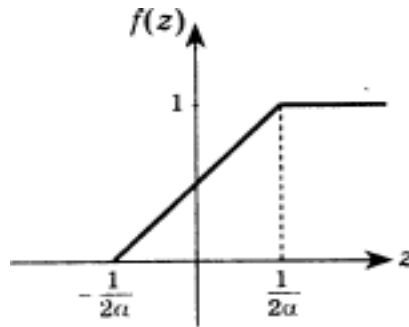


Рисунок 5.4 – Лінійна уніполярна функція з насиченням

Незважаючи на те, що лінійні функції є найбільш простими, їхнє застосування обмежене здебільшого найпростішими ШНМ, які не мають у своєму складі прихованих шарів, у яких, крім того, існує лінійна залежність між вхідними й вихідними змінними. Такі мережі мають обмежені можливості. Двошарова лінійна мережа еквівалентна одношаровій з ваговою матрицею, що дорівнює добутку вагових матриць першого й другого шарів. Звідси випливає, що будь-яка багатошарова лінійна мережа може бути замінена еквівалентною одношаровою. Хоча, використання лінійних активаційних функцій не є зайвим у багатошарових ШНМ, для розширення можливостей мережі застосовують нелінійні функції активації.

У роботі У. Мак-Каллока і У. Піттса як активаційна використовувалася функція Хевісайда – уніполярна порогова гранична функція вигляду

$$f(z) = \begin{cases} 1 & \text{при } z \geq \alpha; \\ 0 & \text{при } z < \alpha. \end{cases}$$

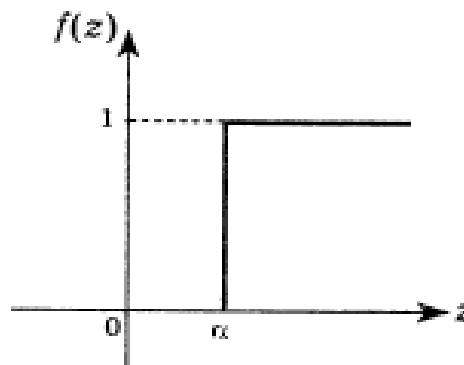


Рисунок 5.5 – Уніполярна порогова функція

Різновидом цієї функції є біполярна порогова функція

$$f(z) = \begin{cases} 1 & \text{при } z \geq \alpha; \\ -1 & \text{при } z < \alpha. \end{cases}$$

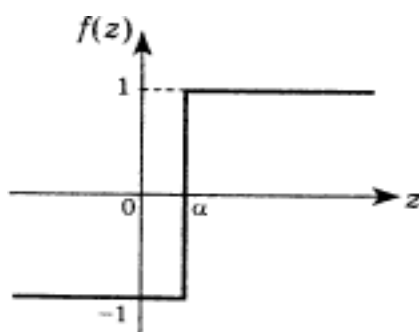


Рисунок 5.6 – Біполярна порогова функція

Ці функції активації застосовувалися переважно в класичних ШНМ. Під час побудови нових структур ШНМ найчастіше доводиться працювати як із самою активаційною функцією, так і з її першою похідною. У цих випадках необхідним є використання як активаційної монотонної диференційованої й обмеженої функції. Особливо важливу роль відіграють такі функції під час моделювання нелінійних залежностей між вхідними й вихідними змінними. Це так звані *логістичні*, або *сигмоїдальні* (S-подібні), функції.

Функція називається сигмоїдальною, якщо вона є монотонно зростаючою, диференційованою і задовольняє умові

$$\lim_{\lambda \rightarrow -\infty} f(\lambda) = k_1, \quad \lim_{\lambda \rightarrow +\infty} f(\lambda) = k_2, \quad k_1 < k_2.$$

До таких функцій належать:

– логістична (уніполярна)

$$f_{log}(z) = \frac{1}{1+e^{-kz}};$$

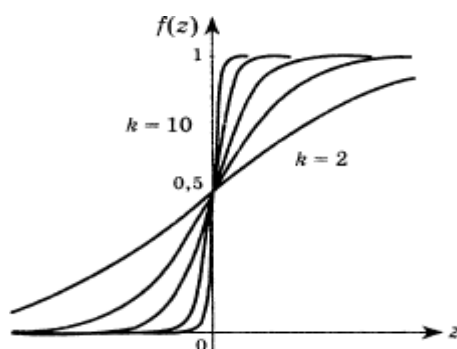


Рисунок 5.7 – Логістична функція

– гіперболічного тангенса (біполярна)

$$f_{th}(z) = \frac{e^{\alpha z} - e^{-\alpha z}}{e^{\alpha z} + e^{-\alpha z}}$$

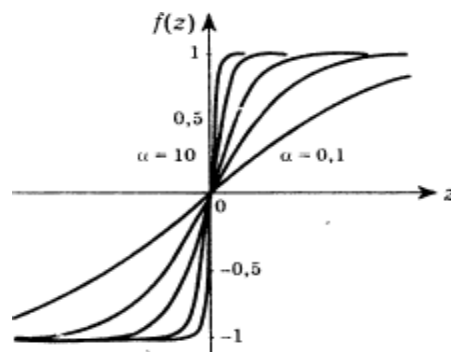


Рисунок 5.8 – Функція гіперболічного тангенса

Моделі штучних нейронів залежать від конкретних застосувань. Тому синтез моделі в кожному окремому випадку є нетривіальним завданням.

### 5.5 Розпізнавання образів. Основні поняття. Математичні моделі.

#### Класифікація та розпізнавання. Поняття образу

*Розпізнаванням* називають процес **розбиття** загальної сукупності об'єктів на групи (класи) зі спільними визначеними параметрами.

*Класифікацією (ідентифікацією)* називають процес визначення класу, до якого відноситься об'єкт їхніми заданими параметрами.

*Ознака* – числова, предикатна, структурна, описова, топологічна характеристика, яку можна використовувати для формального опису об'єкта.

*Простір ознак* – загальна множина всіх можливих значень ознак.

*Образ* – опис одного об'єкта в термінах (за допомогою) ознак; точка у просторі ознак.

*Клас* – підмножина образів, які характеризуються спільними або близькими за значенням ознаками і займають у просторі ознак компакту обмежену область.

*Вектор ознак* – сукупність ознак, за допомогою яких можна відрізнити об'єкти різних класів.

**Система розпізнавання** – це апаратно-програмний комплекс для введення інформації, проведення процедур розпізнавання та класифікації. Процес створення системи розпізнавання складається з декількох **етапів**:

- виявлення числових, предикатних, структурних тощо ознак для опису об'єктів;
- побудова простору ознак;
- проведення попередню процедуру класифікації, аналіз розміщення класів у просторі ознак;
- модифікація, доповнення простору ознак та мінімізація його розміру;
- розробка процедур для створення формальних образів;

– розробка процедур розпізнавання.

Для проведення процедур розпізнавання та класифікації використовуються різні методи та алгоритмічні підходи. Залежно від типу використаних ознак методи розпізнавання поділяються на **дві** загальні групи.

**Паралельні методи** об'єднують алгоритми, у яких використовуються числові та предикатні ознаки.

До **послідовних або структурних методів** відносять системи, де ідентифікація відбувається за описовими, топологічними та структурними ознаками.

Паралельні методи також можна розділити на декілька напрямків, залежно від типу алгоритмів. Тут можна відмітити такі групи, як використання міри, порівняння з еталонами, роздільні функції.

### **Поняття ознаки. Числові, предикатні та структурні ознаки**

*Ознаками* називають числові, предикатні, структурні, описові, топологічні характеристики, які можна використовувати для формального опису об'єкта та його ідентифікації серед інших об'єктів.

Системи розпізнавання, як апаратно-програмні комплекси, можна класифікувати такою схемою (рис. 5.9).

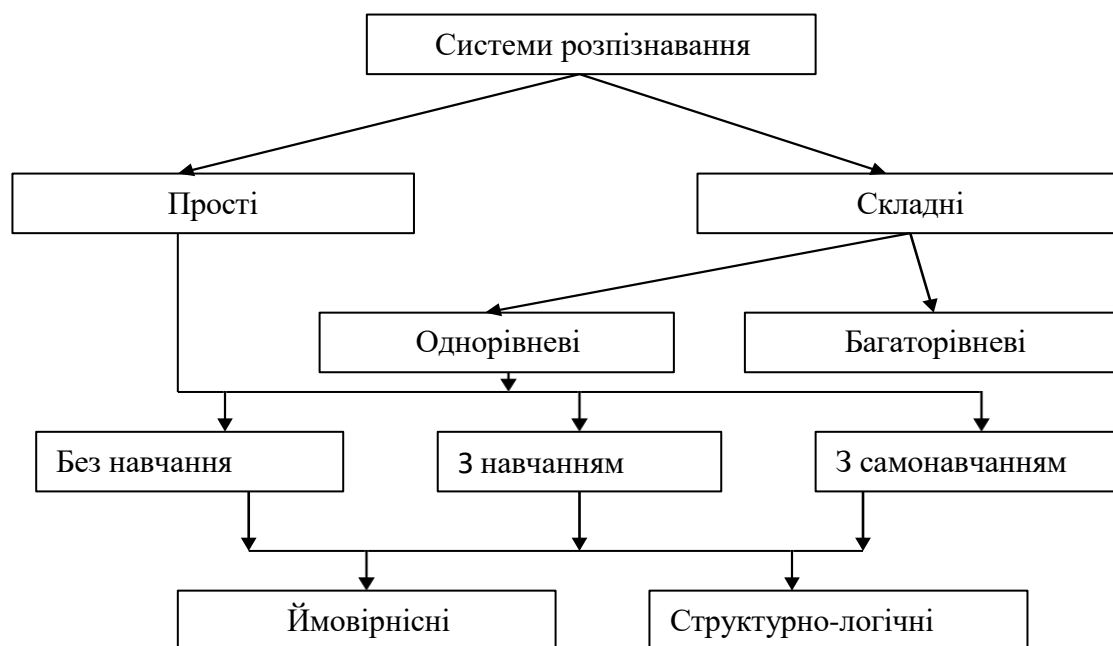


Рисунок 5.9 – Схема систем розпізнавання образів

Числові ознаки визначають деякі кількісні параметри об'єктів. Вони можуть мати чітко визначені значення (кількість лап у комах, кількість копит у ссавців тощо), і тоді ознаку відносять до детермінованого типу, або ознака може приймати випадкові значення в межах деякого діапазону (артеріальний тиск, температура тіла, вага, довжина тощо) і тоді ознаку вважають ймовірнісною.

Предикатні ознаки визначають наявність або відсутність у об'єктів дослідження деяких конкретних характеристик.

До описових ознак відносять вербальні або альтернативні ознаки, такі як колір, відносні розміри, середовище проживання, покриття тіла тощо.

Структурні ознаки визначають склад та взаємне розміщення структурних елементів досліджених об'єктів.

У загальному випадку деяку множину об'єктів можна описувати **змішаною системою** з числових, предикатних, структурних ознак. Якщо опис містить числові ознаки, то за допомогою предикатних, описових та структурних ознак відбувається попередня класифікація об'єктів з метою розбиття загальної множини класів на декілька груп. Числові ознаки використовуються для остаточної ідентифікації в межах групи.

Для сукупності числових ознак будуються відповідні ознакові простори, а опис одного об'єкта сукупністю числових характеристик називають числовим вектором ознак.

За використання числових векторів ознак кожний образ визначається **точкою у просторі** ознак. Оскільки в межах одного класу кожна ознака може отримати фіксовану кількість значень або нескінчену кількість значень з деякого діапазону, можна вважати, що кожний клас **займає** у просторі ознак деяку обмежену область.

Задача розпізнавання або кластеризації **полягає у розбитті загального простору** ознак на окремі області розміщення класів (кластери) та розробки процедур визначення параметрів кожного кластера.

Існують декілька підходів опису або визначення класів у просторі ознак. Залежно від форми представлення класів, у системах розпізнавання застосовуються ті чи інші методи класифікації. У багатьох випадках в задачах кластеризації та класифікації використовується **оцінка (міра) вектору ознак**. Здебільшого, у таких задачах міра пов'язана з поняттям відстані.

У теорії РО [2] використовуються поняття відстаней між точками, від точки до класу, від класу до класу (рисунок 5.10).

**а) Відстань між точками.**

Оскільки розглядаються точки в N-вимірному просторі, можна застосувати наступні відстані:

- Евклідова відстань.

$$d_1(X_1, X_2) = \left\{ \sum_k (x_k^1 - x_k^2)^2 \right\}^{1/2}$$

- Манхетенська відстань.

$$d_2(X_1, X_2) = \sum_k |x_k^1 - x_k^2|.$$

- Чебишевська відстань.

$$d_3(X_1, X_2) = \max_k |x_k^1 - x_k^2|.$$

**б) Відстань від точки до класу.**

$$D(P, X_0) = \inf \{d(P, M), P, M \in R^N, M \in X_0, X_0 \subset R^N\}.$$

**в) Відстань між класами.**

$$D(X_1, X_2) = \inf \{d(P, M), P, M \in R^N, P \in X_1, M \in X_2, X_1, X_2 \subset R^N\}.$$

Рисунок 5.10 = Поняття відстань у теорії розпізнавання образів



Під час використання векторів ознак для розпізнавання та класифікації використовуються такі групи алгоритмів:

- локалізація класів областями прямокутної форми;
- локалізація класів областями сферичної (еліпсоїдальної) форми;
- обчислення мінімальних відстаней;
- використання статистичних (ймовірнісних) характеристик.

### **Контрольні запитання**

1. Дайте визначення нейронної мережі.
2. Що таке нейрон?
3. Назвіть причини використання нейронної мережі.
4. У чому сутність принципів асоціації у нейрокомп'ютерах?
5. На чому базуються моделі штучних нейронів?
6. Опишіть функцію активації.
7. Охарактеризуйте основні поняття теорії розпізнавання.
8. Які системи розпізнавання образів відносяться до детермінованого та ймовірнісного типу?

## **ТЕМА 6 ОСНОВИ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ**

### **6.1 Проблематика нейронних мереж і генетичних алгоритмів**

В останні роки разом із нечіткими системами великий інтерес викликає проблематика нейронних мереж і генетичних алгоритмів. Ці напрямки відносяться до наукової області, обумовленої в англійській літературі терміном «Computational Intelligence».

На рисунку 6.1 видно [8], що задачі нейронних мереж, генетичних алгоритмів і нечітких систем можуть розглядатися поза зв'язком між собою, однак їхня взаємозалежність виявляється надзвичайно важливою. Зокрема, генетичні алгоритми можна застосовувати для підбору ваг і топології нейронної мережі, а також для формування бази правил і функцій приналежності нечіткої системи.

Нейронні мережі також дозволяють вибирати відповідні параметри для самих генетичних алгоритмів (параметри схрещування і мутації); саму філософію нейронних мереж можна закласти у фундамент нечітких систем, що у результаті знаходять здатність до навчання [8].

Крім того, методи теорії нечітких множин дозволяють підбирати як згадані вище параметри генетичних алгоритмів, так і коефіцієнти, що визначають швидкість навчання нейронних мереж. Одним з популярних напрямків Artificial Intelligence є теорія нейронних мереж (neuron nets).

На сьогоднішній день існують **дві** мети нейронного моделювання, що взаємно збагачують одна одну: перша – зрозуміти функціонування нервової системи людини на рівні фізіології і психології і друга – створити обчислювальні

системи (штучні нейронні мережі), що виконують функції, подібні до функцій мозку.



Рисунок 6.1 – Взаємозв’язки між нейронними мережами, генетичними алгоритмами і нечіткими системами

На сьогоднішній день існують **дві** мети нейронного моделювання, що взаємно збагачують одна одну: перша – зрозуміти функціонування нервової системи людини на рівні фізіології і психології і друга – створити обчислювальні системи (штучні нейронні мережі), що виконують функції, подібні до функцій мозку.

Штучні нейронні мережі є моделями нейронної структури мозку, який здатен сприймати, обробляти, зберігати та продукувати інформацію. Особливістю мозку також є навчання та самонавчання на власному досвіді. Адаптивні системи на основі штучних нейронних мереж дозволяють з успіхом вирішувати проблеми розпізнавання образів, виконання прогнозів, оптимізації, асоціативної пам’яті і керування.

Механізм природного мислення базується на збереженні інформації у вигляді образів. Штучні нейронні мережі дозволяють створення паралельних мереж, їх навчання та вирішення інтелектуальних завдань, не використовуючи традиційного програмування. У лексиконі розробників та користувачів нейромереж присутні слова «поводити себе», «реагувати», «самоорганізовувати», «навчати», «узагальнювати» та «забувати».

## 6.2 Модель штучного нейрона

**Штучний нейрон** є базовим модулем нейронних мереж. Він моделює основні функції природного нейрона [8], представленого на рисунку 6.2.

Під час функціонування нейрон одночасно отримує багато вхідних сигналів. Кожен вхід має свою власну синаптичну вагу, яка надає входу вплив, необхідний для функції суматора елемента обробки. Ваги є мірою сили вхідних зв’язків і моделюють різноманітні синаптичні сили біологічних нейронів. Ваги суттєвого входу підсилюються і, навпаки, вага несуттєвого входу примусово зменшується, що визначає інтенсивність вхідного сигналу. Ваги можуть змінюватись відповідно до навчальних прикладів, топології мережі та

навчальних правил. Вхідні сигнали  $x_i$  зважені ваговими коефіцієнтами з'єднання  $w_i$  додаються, проходять через передатну функцію, генерують результат і виводяться.

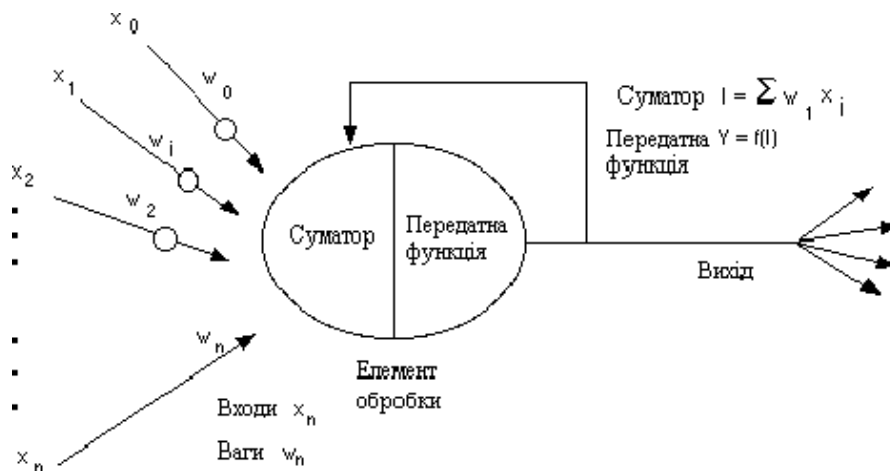


Рисунок 6.2 – Базовий штучний нейрон

У програмних реалізаціях штучні нейрони називають «елементами обробки» або «процесорами» і вкладають в них більше можливостей, ніж у базовому штучному нейроні, що описаний вище.

На рисунку 6.3 зображена детальна схема штучного нейрону [8].

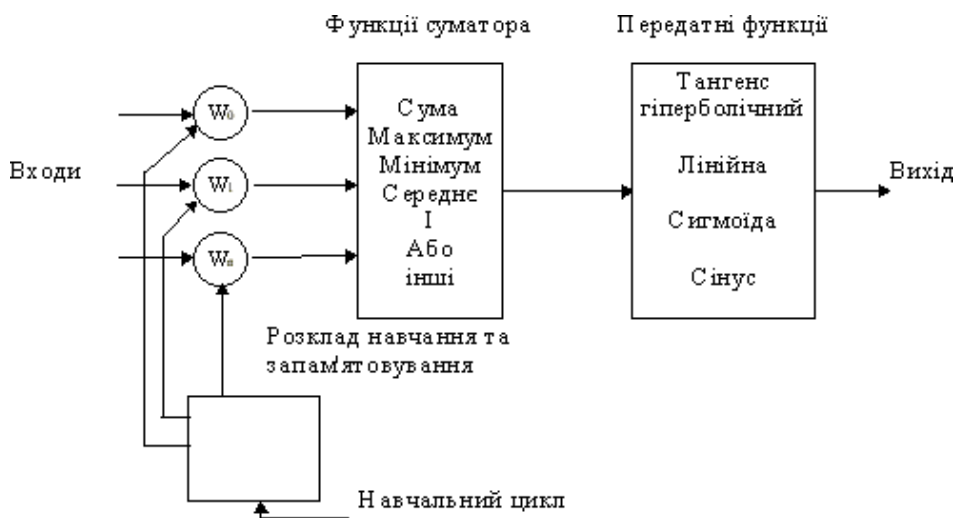


Рисунок 6.3 – Модель «елементу обробки»

Функція суматора може бути складнішою, наприклад, вибір мінімуму, максимуму, середнього арифметичного, добутку або обчислюватися за іншим алгоритмом. Багато програмних реалізацій використовують власні функції суматора, що запрограмовані на мові вищого рівня (C, C++).

Перед надходженням до передатної функції вхідні сигнали та вагові коефіцієнти можуть комбінуватись багатьма способами. Алгоритми для комбінування входів нейронів визначають відповідно до мережної архітектури та парадигми.

У деяких нейромережах суматор виконує додаткову обробку, так звану функцію активації, яка зміщує вихід функції суматора в часі. Цю функцію найкраще використовувати як компоненту мережі в цілому, ніж як компоненту окремого нейрона. Часто, ця функція є відсутньою.

Результат функції суматора перетворюється у вихідний сигнал через передатну функцію. У передатній функції для визначення виходу нейрона загальна сума порівнюється з деяким порогом (зазвичай це діапазон  $[0, 1]$  або  $[-1, 1]$  або інше) за допомогою певного алгоритму.

Переважають застосовують нелінійну передатну функцію, оскільки лінійні (прямолінійні) функції є обмеженими і вихід є пропорційним до входу. Застосування лінійних передатних функцій було проблемою у ранніх моделях мереж, і їхня обмеженість та недоцільність була доведена в книзі Мінські та Пейперта «Перцептрони».

В існуючих нейромережах як передатну функцію використовують сигмоїду, синус, гіперболічний тангенс тощо.

Для різних нейромереж можуть вибиратись інші передатні функції. Після обробки сигналу, нейрон на виході має результат передатної функції, який надходить на входи інших нейронів або до зовнішнього з'єднання, як це передбачається структурою нейромережі.

### **6.3 Архітектура з'єднань штучних нейронів**

**Штучні нейромережі** конструюються з базового блока – **штучного нейрону**. Іншою властивістю нейромереж є величезна кількість зв'язків, які пов'язують окремі нейрони. Групування нейронів у мозку людини забезпечує обробку інформації динамічним, інтерактивним та самоорганізуючим шляхом.

Біологічні нейронні мережі з мікроскопічних компонентів існують у тривимірному просторі і здатні до різноманітних з'єднань. Але для реалізації штучних мереж присутні фізичні обмеження.

Об'єднуючись у мережі, штучні нейрони утворюють систему обробки інформації, яка забезпечує ефективну адаптацію моделі до постійних змін з боку зовнішнього середовища. У процесі функціонування мережі відбувається перетворення вхідного вектора сигналів у вихідний. Конкретний вид перетворення визначається архітектурою нейромережі, характеристиками нейронних елементів, засобами керування та синхронізації інформаційних потоків між нейронами.

**Важливим фактором ефективності мережі** є встановлення оптимальної кількості нейронів та типів зв'язків між ними.

Для опису нейромереж використовують кілька усталених термінів, які в різних джерелах можуть мати різне трактування, зокрема:

**Структура нейромережі** – спосіб зв'язків нейронів у нейромережі.

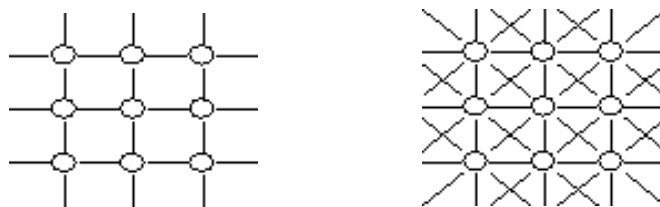
**Архітектура нейромережі** – структура нейромережі та типи нейронів.

**Парадигма нейромережі** – спосіб навчання та використання, іноді містить поняття архітектури.

На базі однієї архітектури може бути реалізовано різні парадигми

нейромережі і навпаки. Серед відомих архітектурних рішень виділяють групу **слабозв'язаних** нейронних мереж, у випадку, коли кожний нейрон мережі зв'язаний лише із сусідніми.

У **повнозв'язаних нейромережах** входи кожного нейрона зв'язані з виходами всіх решти нейронів [8], як це представлено на рисунку 6.4.



а) слабозв'язані нейромережі

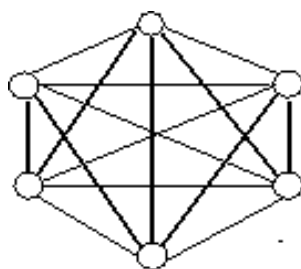


Рисунок 6.4 – Структура нейромережі

Найпоширенішим варіантом архітектури є багат шарові мережі. Нейрони в цьому випадку об'єднуються у прошарки з єдиним вектором вхідних сигналів. Зовнішній вхідний вектор подається на вхідний прошарок нейронної мережі (рецептори). Виходами нейронної мережі є вихідні сигнали останнього прошарку (ефектори). Окрім вхідного та вихідного прошарків, нейромережа має один або кілька прихованих прошарків нейронів [8], які не мають контактів із зовнішнім середовищем (рис.6.5).

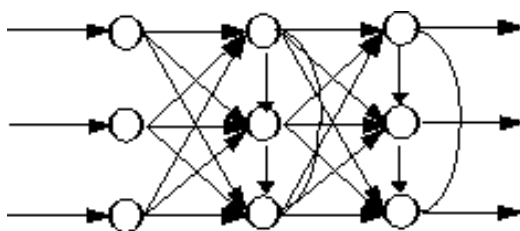


Рисунок 6.5 – Багат шаровий тип з'єднання нейронів

Зв'язки між нейронами різних прошарків називають *проективними*.

Зв'язки між нейронами одного прошарку називають *бічними (латеральними)*.

На рисунку 6.6 показана типова структура штучних нейромереж [8].

Важливим аспектом нейромереж є **напрямок зв'язку** від одного нейрона до іншого.

Зв'язки скеровані від вхідних прошарків до вихідних називаються **аферентними**. Зв'язки в зворотному напрямку називаються **еферентними**.

У більшості мереж кожен нейрон прихованого прошарку отримує сигнали від всіх нейронів попереднього прошарку чи від нейронів вхідного прошарку. Після виконання операцій над сигналами нейрон передає свій вихід до всіх нейронів наступних прошарків, забезпечуючи передачу вперед (feedforward) на вихід.

При зворотному зв'язку вихід нейронів прошарку [8] скеровується до нейронів попереднього прошарку (рис. 6.6).

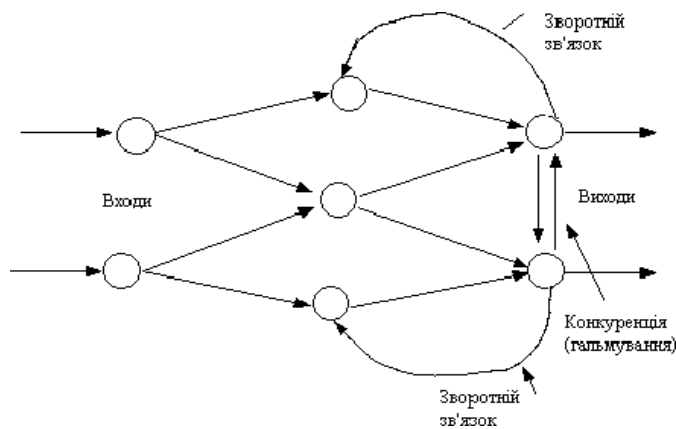


Рисунок 6.6 – Напрямок із зв'язків нейронів

Напрямок зв'язків нейронів значно впливає на роботу мережі. Більшість програмних нейромереж дозволяють користувачу додавати, вилучати та керувати з'єднаннями як завгодно. Корегуючи параметри, можна налаштувати зв'язки як на посилення, так і на послаблення величини сигналів.

За архітектурою зв'язків [8], більшість відомих нейромереж можна згрупувати у два великих класи (рис.6.7):



Рисунок 6.7 – Архітектури нейронних мереж

1. Мережі прямого поширення (з односкерованими послідовними зв'язками).
2. Мережі зворотного поширення (з рекурентними зв'язками).

### Типові архітектури нейронних мереж

Мережі прямого поширення:	Рекурентні мережі:
Перцептрони	Мережа Хопфілда
Мережа Back Propagation	Мережа Хемінга
Мережа зустрічного поширення	Мережа адаптивної резонансної теорії
	Двоскерована асоціативна пам'ять

Мережі *прямого поширення* відносять до *статичних*, тут на входи нейронів надходять вхідні сигнали, які не залежать від попереднього стану мережі.

*Рекурентні мережі вважаються динамічними*, оскільки завдяки зворотним зв'язкам (петлям) входи нейронів модифікуються в часі, що призводить до зміни станів мережі.

#### 6.4 Навчання штучної нейронної мережі

Оригінальність нейромереж як аналога біологічного мозку полягає у здібності до навчання за прикладами, що складають навчальну множину. Процес навчання нейромереж розглядається як налаштування архітектури та вагових коефіцієнтів синаптичних зв'язків відповідно до даних навчальної множини для ефективного вирішення поставленої задачі [4–6].

*Для навчання нейромереж можливо:*

1. Навчання з вчителем (контрольоване навчання).
2. Навчання без вчителя (неконтрольоване навчання).
3. Контрольоване навчання.

Більшість реалізацій нейромереж використовують контрольоване навчання, де вихід, що змінюється, постійно порівнюється з бажаним виходом. Вагові коефіцієнти зв'язків на початку встановлюються випадково (ініціалізація мережі), але під час наступних ітерацій коректуються, щоб досягти близької відповідності між бажаним та біжучим виходами. Такі методи навчання націлені на мінімізацію біжучих похибок всіх елементів обробки, що відбувається завдяки неперервній зміні синаптичних ваг до досягнення прийнятної точності мережі.

Перед використанням, нейромережа з контрольованим навчанням повинна бути навченою. Фаза навчання займає певний час. Навчання вважається закінченим після досягнення нейромережею визначеного користувачем рівня ефективності і бажаної статистичної точності. Після навчання вагові коефіцієнти зв'язків фіксуються для подальшого застосування. Деякі типи мереж дозволяють під час використання продовжувати навчання, і це допомагає мережі адаптуватись до змінних умов.

Навчальні множини повинні бути достатньо великими, щоб містити всю

необхідну інформацію для виявлення важливих особливостей і зв'язків. Навчальні приклади повинні містити широке різноманіття даних. Якщо мережа навчається лише для одного прикладу, вагові коефіцієнти, що старанно встановлено для цього прикладу, радикально змінюються у навчанні для наступного прикладу. Попередні приклади під час навчання наступних просто забуваються. У результаті система повинна навчатись всьому разом, знаходячи найкращі вагові коефіцієнти для загальної множини прикладів.

Наприклад, у навчанні системи розпізнавання піксельних образів для десяти цифр, які подані двадцятьма прикладами кожної цифри, всі приклади цифри «сім» не доцільно подавати послідовно. Краще надати мережі спочатку один тип подання всіх цифр, потім другий тип і так далі.

Головною компонентою для успішної роботи мережі є подання і кодування вхідних і вихідних даних. Штучні мережі працюють лише з числовими вхідними даними, отже, необроблені дані, що надходять із зовнішнього середовища, повинні перетворюватись. Важливою є нормалізація даних, тобто приведення всіх значень даних до єдиного діапазону. Нормалізація виконується шляхом ділення кожної компоненти вхідного вектора на довжину вектора, що перетворює вхідний вектор в одиничний. Попередня обробка зовнішніх даних, отриманих за допомогою сенсорів, у машинний формат є спільною і легко доступною для стандартних комп'ютерів.

Якщо після контрольованого навчання нейромережа ефективно опрацьовує дані навчальної множини, важливим стає її ефективність під час роботи з даними, які не використовувались для навчання. У випадку отримання незадовільних результатів для тестової множини, навчання продовжується. Тестування використовується для забезпечення запам'ятовування не лише даних заданої навчальної множини, але і створення загальних образів, що можуть міститись в даних.

### **Неконтрольоване навчання**

Зараз неконтрольоване навчання використовується в мережах відомих, як самоорганізовані карти (self organizing maps). Мережі не використовують зовнішніх впливів для коректування своїх ваг і внутрішньо контролюють свою ефективність, шукаючи регулярність або тенденції у вхідних сигналах та здійснюють адаптацію відповідно до навчальної функції. Навіть без повідомлення правильності чи неправильності дій мережа повинна мати інформацію відносно власної організації, яка закладена у топологію мережі та навчальні правила.

Алгоритм неконтрольованого навчання скеровано на знаходження близькості між групами нейронів, які працюють разом. Якщо зовнішній сигнал активує будь-який вузол у групі нейронів, дія всієї групи в цілому збільшується. Аналогічно, якщо зовнішній сигнал в групі зменшується, це приводить до гальмівного ефекту на всю групу.

Основу для навчання формує конкуренція між нейронами. Навчання конкуруючих нейронів підсилює відгуки певних груп на певні сигнали. Це пов'язує групи між собою та відгуком. При конкуренції змінюються ваги лише



нейрона-переможця.

Основні етапи розв'язання задач за допомогою нейромереж:

- збір даних для навчання;
- підготовка і нормалізація даних;
- вибір топології мережі;
- експериментальний підбір характеристик мережі;
- експериментальний підбір параметрів навчання;
- власне навчання;
- перевірка адекватності навчання;
- коректування параметрів, остаточне навчання;
- вербалізація мережі з метою подальшого використання.

#### *Вибір топології мережі*

Вибирати тип мережі необхідно виходячи з постановки задачі і наявних даних для навчання. Для навчання з учителем потрібна наявність для кожного елемента вибірки «експертної» оцінки. Іноді одержання такої оцінки для великого масиву даних просто неможливо. У цих випадках природним вибором є мережа Хопфілда без учителя, а також самоорганізуюча карта Кохонена або нейрона. Під час розв'язання таких задач, як прогнозування часових рядів, експертна оцінка вже утримується у вихідних даних і може бути виділена при їхній обробці. У цьому випадку можна використовувати багат шаровий перцептрон або мережу Ворда.

#### *Власне навчання мережі*

У процесі навчання мережа у визначеному порядку переглядає навчальну вибірку. Порядок перегляду може бути послідовним, випадковим і т. д. Мережі, які навчаються без учителя, переглядають вибірку тільки один раз. Під час навчання з учителем мережа переглядає вибірку множини разів, при цьому один повний прохід по вибірці називається епохою навчання.

Зазвичай набір вихідних даних поділяють на дві частини — власне навчальну вибірку і тестові дані; принцип поділу може бути довільним.

Навчальні дані подаються мережі для навчання, а перевірочні використовуються для розрахунку похибки мережі (перевірочні дані ніколи для навчання мережі не застосовуються). Таким чином, якщо на перевірочних даних похибка зменшується, то мережа дійсно виконує узагальнення.

Якщо похибка на навчальних даних продовжує зменшуватися, а похибка на тестових даних збільшується, то виходить, що мережа перестала виконувати узагальнення і просто «запам'ятовує» навчальні дані. Це явище називається перенавчанням мережі або оверфітінгом. У таких випадках навчання зазвичай припиняють. У процесі навчання можуть проявитися інші проблеми, такі як параліч або влучення мережі в локальний мінімум поверхні похибок. Неможливо заздалегідь передбачити прояв тієї або іншої проблеми, так само як і дати однозначні рекомендації до їхнього розв'язання.

#### *Перевірка адекватності навчання*

Навіть у випадку успішного, на перший погляд, навчання мережа не завжди навчається саме тому, чого від неї хотів творець. Відомий випадок, коли

мережа навчалася розпізнаванню зображень танків по фотографіях, однак пізніше з'ясувалося, що всі танки були сфотографовані на тому самому тлі. У результаті мережа «навчилася» розпізнавати цей тип ландшафту, замість того, щоб «навчитися» розпізнавати танки.

Таким чином, мережа «розуміє» не те, що від неї було потрібно, а те, що найпростіше узагальнити.

На рисунку 6.8 подана схема багат шарового перцептрона [8]

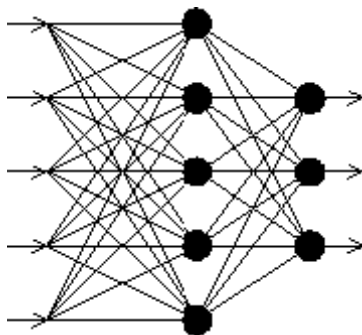


Рисунок 6.8 – Схема багат шарового перцептрона

### 6.5 Мережі Кохонена і Хопфілда

Спеціальний тип нейронної мережі, що дозволяє здійснювати кластеризацію об'єктів. Мережа Кохонена складається з двох прошарків – вхідного і вихідного («прошарок Кохонена») [8]. Кожен нейрон вхідного прошарку пов'язаний зі всіма нейронами вихідного, у нейронів одного прошарку зв'язків немає (рис. 6.9).

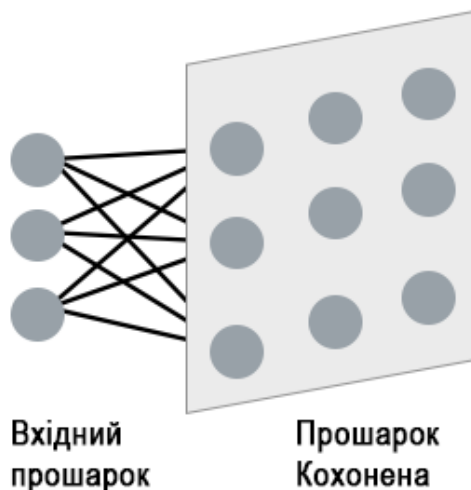


Рисунок 6.9 – Структура мережі Кохонена

Число вхідних нейронів дорівнює кількості ознак об'єкта. Вхідні нейрони не беруть участі в процесі навчання і обробки даних, а просто розподіляють вхідний сигнал по нейронах вихідного шару. Кількість вихідних нейронів мережі Кохонена дорівнює числу кластерів, яке повинно бути побудовано моделлю, і кожен нейрон асоційований з певним кластером.

Ваги на початку встановлюються випадковим чином. Подається перший приклад і обчислюються виходи. Алгоритм навчання діє за принципом «переможець забирає все», тобто нейрону з максимальним значенням виходу привласнюється одиниця, а всім іншим – 0. Після чого об'єкт відноситься до кластеру, асоційованого з даним нейроном-переможцем.

Основний ітераційний алгоритм Кохонена послідовно проходить ряд етапів, на кожному з яких обробляється один приклад з навчальної вибірки. Після обробки достатнього числа прикладів мережа підлаштовує синаптичні ваги під закономірності у вхідних даних та спроможна розподілити схожі приклади до відповідних кластерів. Ваги організуються так, що кожен нейрон вихідного прошарку [8] реагує на відповідну групу схожих вхідних сигналів (рис. 6.10).

У результаті роботи алгоритму центр кластера встановлюється в певній позиції, яка задовольняє схожим прикладам, для яких цей нейрон є «переможцем». У результаті навчання мережі необхідно визначити міру сусідства нейронів, тобто окіл нейрона-переможця, який представляє кілька нейронів, що оточують нейрон-переможець. Спочатку до околу належить велике число нейронів, далі її розмір поступово зменшується. Мережа формує топологічну структуру, у якій на схожі приклади реагують групи нейронів, які близько знаходяться на топологічній карті.

Під час використання навчена мережа Кохонена отримує нові дані і розподіляє їх до відповідних кластерів. Якщо мережа зустрічається з набором даних, несхожим з жодним відомим зразком, вона відносить його до нового кластера або формується відповідь про не розпізнання даних.

Якщо в даних містяться мітки класів, то мережа спроможна вирішувати задачі класифікації. Мережі Кохонена можна використовувати і в задачах, де класи є відомими – перевага буде у спроможності мережі виявляти подібність між різноманітними класами.

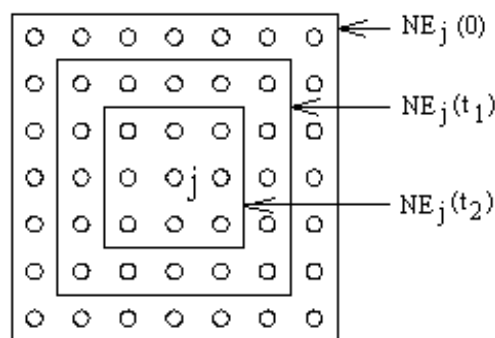


Рисунок 6.10 – Ітераційне визначення нейрона

### Мережа Хопфілда

Нейронна мережа Хопфілда (Hopfield Network, HN) – це повнозв’язна нейронна мережа із симетричною матрицею зв’язків, що реалізує завдання асоціативної пам’яті. Деякий набір двійкових сигналів (зображень, звукових оцифровок і інших даних, що описують якісь об’єкти або характеристики

процесів), вважають зразковим. Мережа повинна вміти з зашумленого сигналу, поданого на її вхід, виділити («пригадати» по частковій інформації) відповідний зразок або «дати висновок» про те, що вхідні дані не відповідають жодному із зразків.

Мережа Хопфілда використовує **три** прошарки: вхідний, прошарок Хопфілда та вихідний прошарок. Кожен прошарок має однакову кількість нейронів. Виходи нейронів вхідного прошарку надходять до входів відповідних нейронів прошарку Хопфілда. Зв'язки мають фіксовані вагові коефіцієнти. Виходи прошарку Хопфілда під'єднуються до входів всіх нейронів прошарку Хопфілда, за винятком самого себе, а також до відповідних елементів у вихідному прошарку. Під час навчання мережа скеровує дані з вхідного прошарку до прошарку Хопфілда. Прошарок Хопфілда коливається [8], поки не буде завершена певна кількість циклів, і біжучий стан сигналів нейронів прошарку передається на вихідний прошарок (рис.6.11). Цей стан відповідає образу, який буде запам'ятовано в мережі.

Навчання мережі Хопфілда вимагає, щоб навчальний образ був поданий на вхідному та вихідному прошарках одночасно. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань. Для правильного навчання мережі відповідні пари «вхід-вихід» мають відрізнитися між собою.

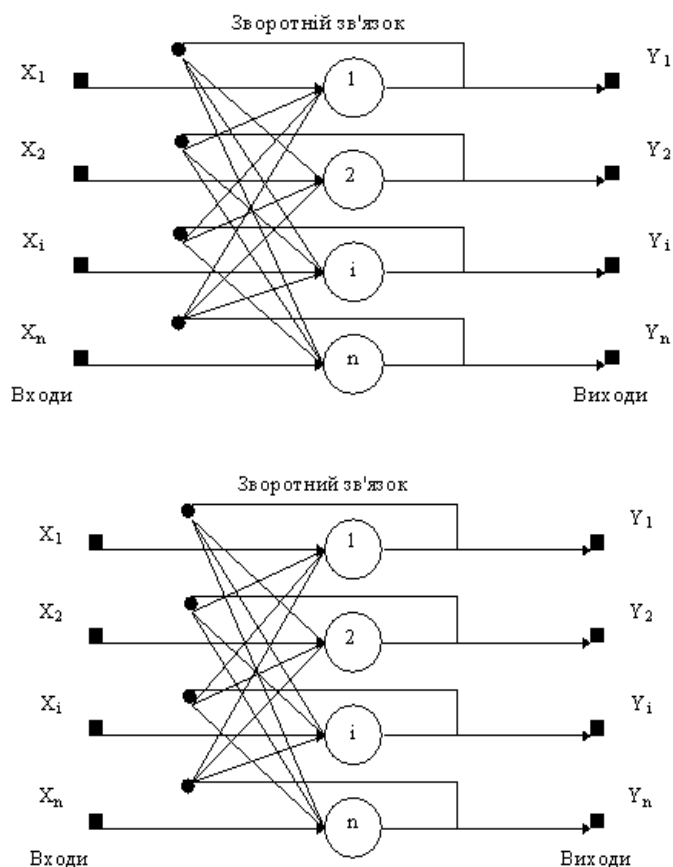


Рисунок 6.11 – Ітераційне визначення нейрона-переможця

Якщо мережа Хопфілда використовується як пам'ять, що адресується за змістом, вона має два головних обмеження.

1. Число образів, що можна зберегти та точно відтворити, є строго обмеженим. Якщо зберігається занадто багато образів, мережа може збігтись до нового неіснуючого образу, відмінному від всіх запрограмованих образів, або не збігтись взагалі. Межа ємності пам'яті для мережі приблизно 15 % від числа нейронів у прошарку Хопфілда.

2. Якщо навчальні приклади є занадто подібними, прошарок Хопфілда може стати нестабільним. Зразок образу вважається нестабільним, якщо він застосовується за нульовий час і мережа збігається до деякого іншого образу з навчальної множини. Ця проблема може бути вирішена вибором навчальних прикладів, що достатньо відрізняються між собою.

### Контрольні запитання

1. Опишіть взаємозв'язки між нейронними мережами, генетичними алгоритмами і нечіткими системами.

2. Проведіть класифікацію нейронних мереж за типом навчання. Наведіть приклади.

3. Назвіть базові типи нейромереж за напрямком потоку обробки даних. У чому полягає різниця?

4. Поясніть принцип роботи простого перцептрона, які завдання він спроможний вирішити.

5. Назвіть особливість алгоритму навчання з учителем.

6. Назвіть загальні правила формування структури нейромережі щодо кількості шарів та нейронів.

7. Які особливості закладено в мережі Кохонена? Для яких завдань ці мережі є ефективними?

8. Назвіть особливості, тип і призначення мережі Хопфілда.

9. Яку основну ідею покладено в архітектуру нейромережі?

## ТЕМА 7 ЕКСПЕРТНІ СИСТЕМИ – СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ

### 7.1 Основа експертної системи

**Експертна система** (далі – ЕС) – це система штучного інтелекту, що використовує накопичені знання для забезпечення високоєфективного рішення задач у вузькій професійній діяльності.

**База знань** може лягти в основу експертної системи, тобто інформаційної системи, яка здійснює дедуктивне виведення на основі наявних знань. Існують різні визначення експертних систем, основна суть яких зводиться до такого.

**Експертні системи** – це інтелектуальні програмні засоби, здатні у діалозу з людиною одержувати, накопичувати та коригувати знання із заданої

предметної галузі, виводити нові знання, розв'язувати на основі цих знань практичні задачі та пояснювати хід їхнього розв'язку.

Експертні системи акумулюють знання експертів — провідних спеціалістів у цій предметній галузі. В основі роботи експертних систем лежить дедуктивне виведення нових тверджень з існуючих. Типове застосування експертних систем – консультації для фахівців середньої кваліфікації і неспеціалістів у тій галузі, для якої вона розроблена.

Створити універсальну експертну систему неможливо. По-перше, це пов'язано з «прокляттям розмірності». По-друге, експертна система повинна акумулювати знання людей-експертів, а «універсальних» експертів не існує і не може існувати. По-третьє, жодне логічне виведення не може замінити інтуїцію та досвід експерта [1, 4, 8].

Можна створити тільки експертні системи, які належать до конкретної предметної галузі. Остання передбачає лімітований набір явищ і понять певної сфери людської діяльності та обмежене коло задач, які вирішуються у цій галузі. Існує чимало експертних систем у таких сферах, як медична і технічна діагностика, пошук корисних копалин, юриспруденція, аналіз інвестицій і комерційних ризиків і т. п. У створенні експертних систем повинні брати участь фахівці як мінімум двох категорій:

- експерт, що є висококваліфікованим фахівцем у цій предметній області, знання якого потрібно передати експертній системі;
- інженер знань, завдання якого – формалізувати знання експерта і привести їх до вигляду, придатного для занесення до бази знань.

Серйозна проблема у цьому випадку пов'язана з тим, що знання експертів важко формалізувати. Експерт часто не може сформулювати свої знання у явному вигляді, робить правильні висновки, але *не може пояснити, як саме він їх робить*. Якщо ж експерт і формулює певні правила виведення, вони далеко не завжди відзначаються точністю та адекватністю. З цього випливає ряд інших труднощів, які так чи інакше виявляють себе на етапі формалізації знань або застосування готових експертних систем.

## **7.2 Технологія побудови експертної системи**

Експертна система – комп'ютерна програма, здатна частково замінити фахівця-експерта у вирішенні проблемної ситуації. Це обчислювальна система, у яку включені знання фахівців про деяку вузьку предметну область у формі бази знань. Такі системи можуть використовуватися експертом для поліпшення здатностей і можливостей у рішенні завдань певного класу в конкретній предметній області. ЕС можуть бути використані для поширення джерел рідких знань. Ці системи можуть мати значний вплив як на діяльність таких професійних консультантів, як фінансові аналітики, юристи, аудиторы й ін., так і на організації і їхній менеджмент.

У середині експертної системи немає заздалегідь заданого дерева питань, кожне наступне питання вибирається виходячи з відповідей на всі попередні. Це дозволяє виключити зайві питання й не видавати варіанти відповіді, які не приведуть до яких-небудь результатів. Відсутність фіксованого дерева дозволяє

користувачеві задавати пріоритет питань, вибираючи найбільш важливі для себе аспекти в процесі пошуку. У будь-який момент можна знову повернутися до питання й вибрати іншу відповідь без необхідності знову відповідати на інші питання.

Експертні системи мають одну велику відмінність від інших ПС: вони не призначені для рішення якихось універсальних завдань, як наприклад, нейронні мережі або генетичні алгоритми. Експертні системи [4] призначені для якісного рішення завдань у певної розроблювачами області, у рідких випадках – областях (рис.7.1).

**Характерними рисами ЕС є:**

- чітка обмеженість предметної області;
- здатність ухвалювати рішення в умовах невизначеності;
- здатність пояснювати хід і результат рішення зрозумілим для користувача способом;
- чіткий поділ декларативних і процедурних знань (фактів і механізмів виводу);
- здатність поповнювати базу знань, можливість нарощування системи;
- результат видається у вигляді конкретних рекомендацій для дій у сформованій ситуації, що не уступають рішенням кращих фахівців;
- орієнтація на рішення неформалізованих (спосіб формалізації поки невідомий) завдань;
- алгоритм рішення не описується заздалегідь, а будується самою експертною системою;
- відсутність гарантії знаходження оптимального рішення з можливістю вчитися на помилках.

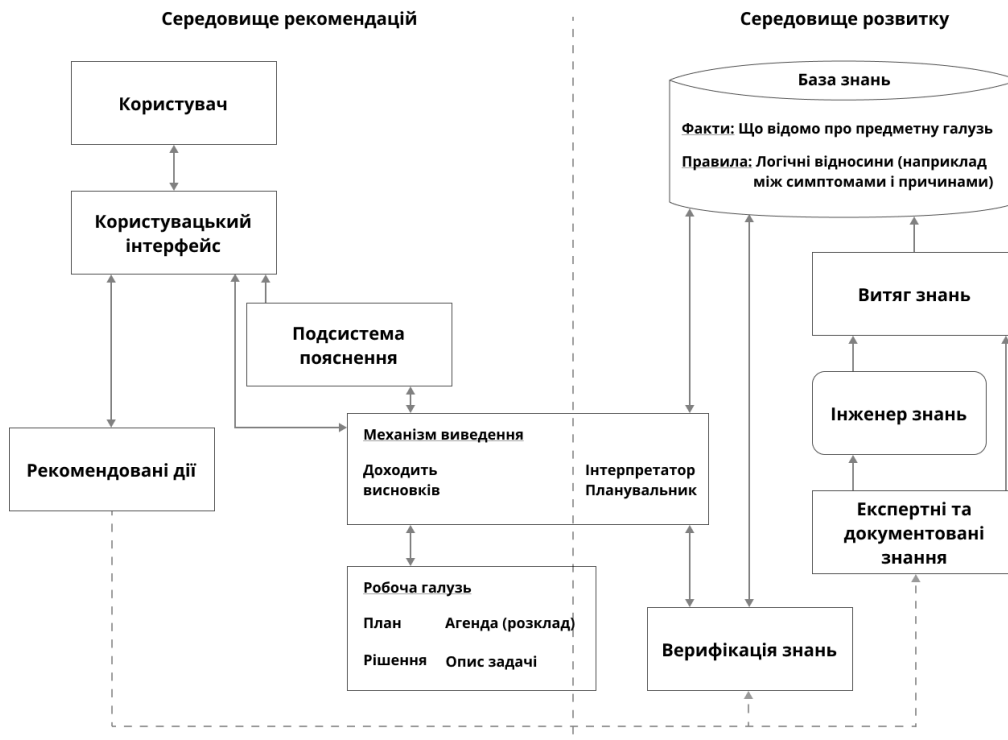


Рисунок 7.1 – Експертна система

На схемі 7.2 представлена послідовність побудови експертної системи [4].

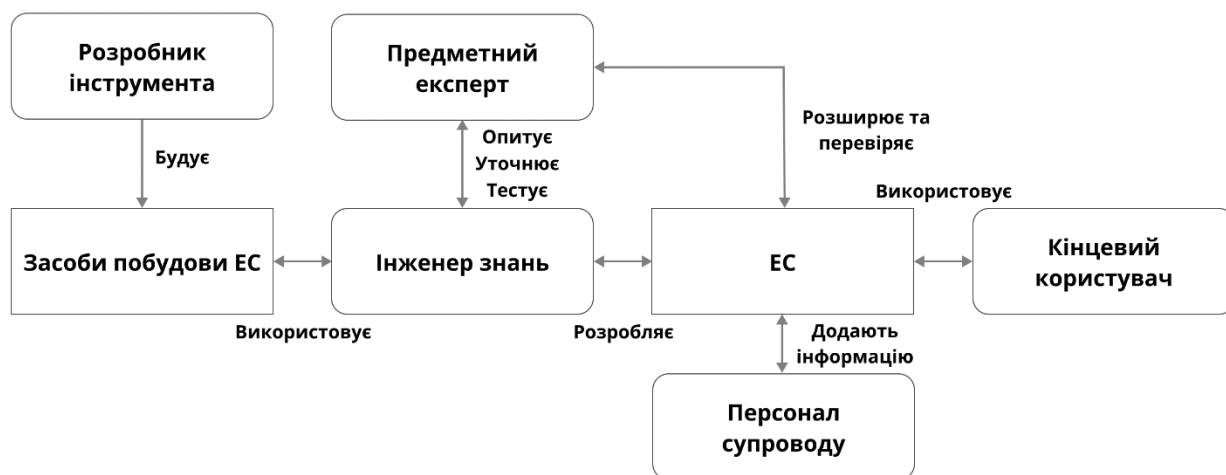


Рисунок 7.2 – Процес побудови ЕС

### Класифікації ЕС:

1. Властиво експертні системи.
2. Інтерактивні баннери (web + ЕС).
3. Інтерактивні мовці баннери ЕС – це інфи або експертні системи, призначені для розміщення на зовнішніх ресурсах.

### Переваги інтерактивних баннерів:

1. Підвищена привабливість для споживачів – з незвичайним баннером хочеться поспілкуватися.
2. Тривалий контакт із користувачем. Середній час спілкування з баннером може становити близько 3 хвилин.
3. Баннер може вести різних співрозмовників на різні сторінки, відповідно до їхніх запитів і потреб.

### Класифікація ЕС за зв'язком із реальним часом:

1. Статичні ЕС – це ЕС, що вирішують завдання в умовах не вихідних даних, що змінюються в часі, і знань.
2. Квазидинамічні ЕС інтерпретують ситуацію, що міняється з деяким фіксованим інтервалом часу.
3. Динамічні ЕС – це ЕС, що вирішують завдання в умовах вихідних даних, що змінюються в часі, і знань.

### 7.3 Структура експертної системи

На рисунку 7.3. подана канонічна структура експертної системи динамічного типу [4].

Ця система має такі елементи.

- механізм логічного виводу, називаний також інтерпретатором, вирішувачем;
- робочу пам'ять (РП), яку називають також робочою базою даних (БД);
- базу знань (БЗ);
- підсистему придбання й поповнення знань;
- підсистему пояснення;



- підсистему діалогу;
- підсистему взаємодії із зовнішнім світом.

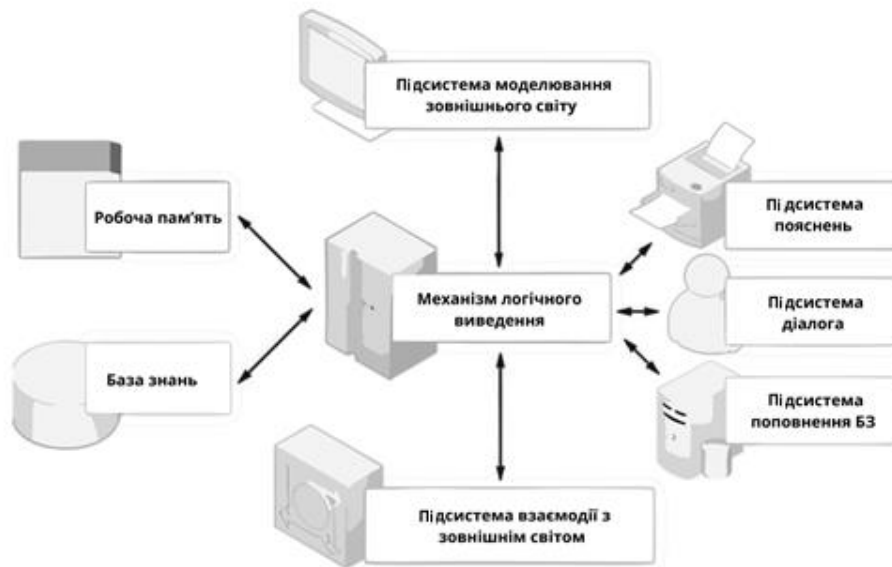


Рисунок 7.3 – Канонічна структура ЕС

Механізм логічного виводу (далі – МЛВ) призначений для одержання нових фактів на основі зіставлення вихідних даних з робочої пам'яті й знань із бази знань. Механізм логічного виводу реалізує алгоритми прямого й/або зворотного виводу й формально може бути представлений четвіркою.

Механізм виводу є мозком ЕС, його також називають керуюча структура або інтерпретатор правил (в ЕС, заснованих на правилах).

Цей компонент є переважно комп'ютерною програмою, що забезпечує методологію для міркування про інформацію в БЗ і в робочій області, а також для формулювання висновків. Вона забезпечує вказівки про те, як використовувати знання системи при реалізації оренди (розкладу запланованих дій у робочій області), що організує й управляє кроками, що вживаються для вирішення завдання.

Механізм виводу має два головних елементи:

- інтерпретатор, що виконує обрані позиції оренди, використовуючи відповідні правила БЗ;
- планувальник, що підтримує керування **агендой**. Він оцінює результати використовуваних правил виводу у світлі їхніх пріоритетів або інших критеріїв в агенде.

Робоча пам'ять призначена для зберігання вихідних і проміжних фактів розв'язуваного в сучасний момент завдання, зазвичай розміщується в оперативній пам'яті ЕОМ і відбиває поточний стан предметної області у вигляді фактів з коефіцієнтами упевненості (далі – КУ) в істинності цих фактів.

Цінність всієї експертної системи як закінченого продукту на 90 % визначається якістю створеної бази знань. БЗ ЕС зазвичай містить факти

(статичні відомості про предметну область) і правила – набір інструкцій, застосовуючи які до відомих фактів, можна одержувати нові факти. У рамках логічної моделі баз даних і бази знань записуються мовою Пролог (мова й система логічного програмування) за допомогою мови предикатів для опису фактів і правил логічного виводу, що виражають правила визначення понять, для опису узагальнених і конкретних відомостей, а також конкретних і узагальнених запитів до баз даних і баз знань.

Підсистема придбання й поповнення знань автоматизує процес наповнення експертної системи знаннями, здійснюваний користувачем-експертом, і адаптації бази знань системи до умов її функціонування. Адаптація експертної системи до змін у предметній області реалізується шляхом заміни правил або фактів у базі знань.

Підсистема пояснення пояснює, як система одержала рішення завдання (або чому вона не одержала рішення) і які знання вона при цьому використовувала, що полегшує експертові тестування системи й підвищує довіру користувача до отриманого результату. Можливість пояснювати свої дії є одним з найважливіших властивостей експертної системи, тому що:

- підвищується довіра користувачів до отриманих результатів;
- полегшується налагодження системи;
- створюються умови для користувачів по розкриттю нових закономірностей предметної області;
- пояснення отриманих висновків може слугувати засобом пошуку точки в парето-оптимальній множині рішень.

У цей час на практиці все СО реалізуються на тих самих принципах переважно двома способами:

- фіксацією подій і станів за допомогою заготовлених текстів природною мовою;
- трасуванням міркувань, зворотним розгортанням дерева цілей із вказівкою підцілей.

При реалізації кожного із цих способів попередньо виділяються ситуації, факти й вузли переходу в нові стани, що вимагають пояснень. Їм ставиться у відповідність деякий текст пояснення.

Структура експертної системи була б неповною без підсистеми діалогу. Підсистема діалогу орієнтована на організацію дружнього інтерфейсу з усіма категоріями користувачів як у ході рішення завдань, так і в ході придбання знань і пояснення результатів роботи.

Факти й правила в експертній системі не завжди або істинні, або помилкові. Іноді існує деякий ступінь непевності у вірогідності факту або точності правила. Якщо цей сумнів виражений явно, то воно називається «коефіцієнтом довіри».

Коефіцієнт довіри – це число, що означає ймовірність або ступінь упевненості, з якої можна вважати цей факт або правило достовірним, або справедливим. Цей коефіцієнт є оцінкою ступеня довіри до рішення, виданого експертною системою.

### Режими функціонування ЕС передбачає:

- режим уведення знань – у цьому режимі експерт за допомогою інженера по знаннях і за допомогою редактора бази знань вводить відомі йому відомості про предметну область у базу знань ЕС;
- режим консультації – користувач веде діалог з ЕС, повідомляючи її відомості про поточне завдання й одержуючи рекомендації ЕС. Наприклад, на основі відомостей про фізичний стан хворого ЕС ставить діагноз у вигляді переліку захворювань, найбільш імовірних при даних симптомах.

Таблиця 7.1 – Основні класи рішення завдань, які розв’язують ЕС

Клас	На вирішення якого завдання спрямована
Інтерпретація	Виявлення описів ситуації зі спостережень
Пророкування	Виявлення схожих наслідків у такій ситуації
Діагностика	Виявлення несправності системи через спостереження
Проектування	Конфігурування й розробка об’єктів, що задовольняють певним вимогам
Планування	Розробка планів для досягнення цілей
Моніторинг	Порівняння спостережень із планами, сигналізуючи про відхилення й виключення
Налагодження	Виявлення й усунення несправностей
Управління	Інтерпретування, пророкування, відновлення й моніторинг поведінки системи

Деякі ЕС належать до двох або більше із цих категорій. Дамо короткий опис кожних категорій.

Системи інтерпретації виявляють описи ситуації зі спостережень. Ця категорія містить спостереження, розуміння мови, аналіз образів, інтерпретацію сигналів і багато інших видів інтелектуального аналізу.

Системи інтерпретації пояснюють спостережувані дані шляхом присвоєння їм символічних значень, що описують ситуацію.

Системи пророкування містять прогнозування погоди, демографічні пророкування, економічне прогнозування, оцінки врожайності, а також військове, маркетингове й фінансове прогнозування.

Системи діагностики містять діагностику в медицині, електроніці, механіку й програмному забезпеченні. Діагностуючі системи зазвичай співвідносять спостережувані поведінкові відхилення із причинами, що лежать в основі.

Системи проектування розробляють конфігурації об’єктів, які задовольняють певним вимогам завдання проектування. Такі завдання включають конструювання будинків, планування розташування встаткування й ін. Ці системи конструюють різні взаємозв’язки описів об’єктів один з одним і

перевіряють, чи задовольняють ці конфігурації встановленим обмеженням і вимогам.

Системи планування спеціалізуються на завданнях планування, наприклад, таких як автоматичне програмування. Вони також працюють із коротко й довгостроковим плануванням у керуванні проєктами, маршрутизацією, комунікацією, розробкою продукту, військовими додатками, виробничим й фінансовим плануванням.

Системи моніторингу порівнюють спостереження поведження системи зі стандартами, які представляються визначальними для досягнення мети. Ці вирішальні виявлення відповідають потенційним недолікам на підприємстві. Існує багато комп'ютерних систем моніторингу: від контролю руху повітряних потоків до завдань керування збором податків.

Системи керування й контролю адаптивно управляють загальним поведженням системи. Для здійснення цього система керування повинна періодично інтерпретувати поточну ситуацію, пророкувати майбутнє, діагностувати причини очікуваних проблем, формулювати план усунення цих проблем і здійснювати моніторинг його виконання для забезпечення успіху.

Найбільш відомі / розповсюджені ЕС:

- CLIPS – популярна ЕС (public domain);
- OpenCus – потужна динамічна ЕС із глобальною онтологічною моделлю й підтримкою незалежних контекстів;
- MYCIN – найбільш відома діагностична система, що призначена для діагностики й спостереження за станом хворого при менінгіті й бактеріальних інфекціях;
- HASP/SIAP – інтерпретуюча система, що визначає місце розташування й типи судів у Тихому океані за даними акустичних систем спостереження.

Першу ЕС за назвою «Dendral» розробили в Стенфорді наприкінці 1960-х рр. Вона визначала будову органічних молекул за хімічними формулами і спектрографічними даними про хімічні зв'язки в молекулах. Цінність Dendral полягала в такому: органічні молекули зазвичай дуже великі й тому число можливих структур цих молекул велике. Завдяки евристичним знанням експертів-хіміків, закладених в ЕС, правильне рішення з мільйона можливих обиралося за кілька спроб [8].

Принципи й ідеї, закладені в Dendral, виявилися настільки ефективними, що вони дотепер застосовуються в хімічних і фармацевтичних лабораторіях по всьому світі. ЕС Dendral однією з перших використовувала евристичні знання фахівців для досягнення рівня експерта у вирішенні завдань, однак методика сучасних експертних систем пов'язана з іншою розробкою – Musin. У ній використовувалися знання експертів медицини для діагностики й лікування спеціального менінгіту й бактеріальних інфекцій крові. ЕС Musin, також розроблена в Стенфорді в середині 1970-х рр., однією з перших звернулася до проблеми ухвалення рішень на основі ненадійної або недостатньої інформації.

Всі міркування експертної системи Musin були засновані на принципах керуючої логіки, що відповідають специфіці предметної області. Багато методик

розробки експертних систем, що використовуються сьогодні, були вперше розроблені в межах проєкту Mycin. MYCIN була ранньою експертною системою, розробленою за 5 або 6 років на початку 1970-х років у Стендфордському університеті.

Вона була написана на мові ЛІПС як докторська дисертація Edward Shortliffe під керівництвом Bruce Buchanan, Stanley N. Cohen і інших. У цій же лабораторії була раніше створена експертна система Dendral, але цього разу увага була акцентована на використанні вирішальних правил з елементами невизначеності. MYCIN був спроектований для діагностування бактерій, що викликають важкі інфекції. Назва системи походить від суфікса «-мицин», що часто зустрічається в назвах антибіотиків. Також Mycin використовувалася для діагностики захворювань згортання крові.

#### **Переваги ЕС:**

##### **1. Сталість.**

Людська компетенція слабшає згодом. Перерва в діяльності людини-експерта може серйозно відбитися на його професійних якостях.

##### **2. Легкість передачі.**

Передача знань від однієї людини іншій – довгий і дорогий процес. Передача штучної інформації – це простий процес копіювання програми або файлу даних.

##### **3. Стійкість і відтворюваність результатів.**

Експертні системи стійкі до «перешкод». Людина ж легко піддається впливу зовнішніх факторів, які безпосередньо не пов'язані з розв'язуванням завданням. Експерт-людина може ухвалювати в тотожних ситуаціях різні рішення через емоційних факторів. Результати експертної системи є стабільними.

##### **4. Вартість.**

Експерти, особливо висококваліфіковані, обходяться дуже дорого. Експертні системи, навпаки, порівняно недорогі. Їхня розробка дорога, але вони дешеві в експлуатації.

Крім того, експерт-людина може ухвалювати різні рішення в тотожних ситуаціях через емоційних факторів (вплив дефіциту часу, вплив стресу).

Таблиця 7.2 – Порівняння людської й штучної компетентності

<b>Людська компетентність</b>	<b>Штучна компетентність</b>
Неміцна	Постійна
представляється Важко	Легко передана
Важко документована	Легко документована
Непередбачена	Стійка
Дорога	Прийнятна по витратах

## Недоліки ЕС

На сьогоднішні створена вже велика кількість експертних систем. За допомогою їх вирішується широке коло завдань, але винятково в вузько спеціалізованих предметних областях. Зараз розвиток експертних систем трохи призупинився, і цьому є низка причин:

- передача експертним системам «глибоких» знань про предметну область є великою проблемою. Це зазвичай є наслідком складності формалізації евристичних знань експертів;

- експертні системи нездатні надати осмислені пояснення своїх міркувань, як це робить людина. Експертні системи всього лише описують послідовність кроків, початих у процесі пошуку рішення;

- налагодження й тестування будь-якої комп'ютерної програми є доволі трудомісткою справою, але перевіряти експертні системи особливо важко. Це є серйозною проблемою, оскільки експертні системи застосовуються в таких критичних областях, як керування повітряним і залізничним рухом, системами зброї й у ядерній промисловості;

- експертні системи мають ще один більший недолік: вони нездатні до самонавчання. Для того щоб підтримувати експертні системи в актуальному стані, необхідно постійне втручання в базу знань інженерів із знань. Експертні системи, позбавлені підтримки з боку розроблювачів, швидко втрачають свою затребуваність;

- експерти можуть безпосередньо сприймати комплекс вхідної сенсорної інформації (візуальної, звукової, дотикальної, нюхової й тактильної). ЕС – тільки символи. Хоча в окремих напрямках розробки інженерних і виробничих інтелектуальних систем отримані реальні результати певної обробки сенсорної інформації;

- експерти – люди можуть охопити картину в цілому, всі аспекти проблеми й зрозуміти, як вони співвідносяться з основним завданням. ЕС прагне зосередити на самому завданні, хоча суміжні завдання можуть вплинути на рішення основної;

- люди, експерти й не експерти, мають те, що ми називаємо здоровим глуздом, або загальнодоступними знаннями. Це широкий спектр загальних знань про світ, про те, які закони в ньому діють, тобто знання, якими кожний з нас володіє, здобуває з досвіду і якими постійно користується. Через величезний обсяг знань, що утворюють здоровий глузд, не існує легкого способу вмонтувати їх в інтелектуальну програму. Знання здорового глузду включають знання про те, що ви знаєте й чого не знаєте.

Тому ЕС найчастіше використовують як порадників, як консультантів або помічників ОПР (особи, що приймає рішення).

ВР легко піддаються навчанню й допомагають вирішити безліч завдань, що постають перед замовником. Вони можуть бути:

- консультантами, що відповідають на питання користувачів про представлені товари й послуги;

- продавцями, що допомагають підібрати потрібний товар, послугу, тариф і т. п.;
- співробітниками технічної підтримки, що допомагають користувачеві вирішити виниклі технічні проблеми;
- промоутерами, що просувають нові товари й послуги;
- цікавими співрозмовниками, що викликають інтерес, що підвищують настрій і лояльність відвідувачів.

#### **Сфери застосування ЕС:**

- банки й страхові компанії, яким важливо мати на сайті грамотного консультанта, здатного оперативно розповісти всі подробиці про надавані послуги;
- інтернет-магазини, яким важливо допомагати клієнтам у виборі товарів, а також просувати акції й розпродажі;
- інтернет-портали, яким необхідно привертати увагу користувачів до їхніх внутрішніх проєктів;
- організатори заходів, яким важливо інформувати відвідувачів сайту про всі новини й подробиці;
- компанії, що роблять технічні послуги, яким важливо забезпечити цілодобову технічну підтримку користувачів.

**Забезпечення роботи ПС** – математичне, лінгвістичне, програмне, технічне, технологічне, кадрове.

У загальному випадку всі системи, засновані на знаннях, можна поділити на системи, що розв'язують задачі аналізу, і на системи, які розв'язують задачі синтезу. Основна відмінність задач аналізу від задач синтезу полягає в тому, що якщо в задачах аналізу множина рішень може бути перерахована і включена в систему, то в задачах синтезу множина рішень потенційно необмежена. Задачею аналізу є: інтерпретація даних, діагностика, підтримка ухвалення рішення; до завдань синтезу відносять проєктування, планування, управління. Комбіновані задачі: навчання, моніторинг, прогнозування.

### **Контрольні запитання**

1. Дайте визначення «експертним системам».
2. Опишіть, для чого призначені експертні системи?
3. Опишіть структуру експертних систем.
4. Назвіть переваги і недоліки експертних систем.
- 5 Назвіть основні класи рішення завдань, які розв'язують ЕС.

## СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Климчук О. В. Інформаційні системи і технології в управлінні : конспект лекцій / О. В. Климчук. – Вінниця : ДонНУ імені Василя Стуса, 2021. – 160 с.
2. Коцовський В. М. Інтелектуальні інформаційні системи : конспект лекцій / В. М. Коцовський. – Ужгород : ДВНЗ «Ужгородський національний університет», 2019. – 73 с.
3. Шаров С. В. Інтелектуальні інформаційні системи : навч. посіб. / С. В. Шаров, Д. В. Лубко, В. В. Осадчий. – Мелітополь : Вид-во МДПУ ім. Б. Хмельницького, 2015. – 144 с.
4. Нестеренко О. В. Інтелектуальні системи і технології : навч. посіб. / О. В. Нестеренко, О. В. Ковтунець, О. О. Фаловський. – Київ : Національна академія управління, 2017. – 90 с.
5. Субботін С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень / С. О. Субботін. – Запоріжжя : ЗНТУ, 2008. – 341 с.
6. Вахнюк С. В. Технологія створення програмних та інтелектуальних систем : навч. посіб. / С. В. Вахнюк. – Суми : ДВНЗ «УАБС НБУ», 2011. – 254 с.
7. Недашківський О. Л. Планування та проектування інформаційних систем : навч. посіб. / О. Л. Недашківський. – Київ, 2014. – 215 с.
8. Терейковський І. А. Штучні нейронні мережі : базові положення / І. А. Терейковський, Д. А. Бушуєв, Л. О. Терейковська. – Київ : КПІ ім. Ігоря Сікорського, 2022. – 123 с.



*Електронне навчальне видання*

**СІЗОВА** Наталія Дмитрівна

## **ІНТЕЛЕКТУАЛЬНІ УПРАВЛЯЮЧІ СИСТЕМИ І ТЕХНОЛОГІЇ**

### **КОНСПЕКТ ЛЕКЦІЙ**

*(для здобувачів другого (магістерського) рівня вищої освіти денної,  
заочної і дистанційної форм навчання  
зі спеціальності 122 – Комп'ютерні науки)*

Відповідальний за випуск *М. В. Новожилова*

Редактор *О. В. Михаленко*

Комп'ютерне верстання *Н. Д. Сізова*

План 2024, поз. 112Л

---

Підп. до друку 24.04.2024. Формат 60 × 84/16.

Ум. друк. арк. 3,9

Видавець і виготовлювач:

Харківський національний університет  
міського господарства імені О. М. Бекетова,

вул. Маршала Бажанова, 17, Харків, 61002.

Електронна адреса: [office@kname.edu.ua](mailto:office@kname.edu.ua)

Свідоцтво суб'єкта видавничої справи:

ДК № 5328 від 11.04.2017.