

***БАКАЛАВРСКА
РОБОТА***

БР.ІІ – 05.00.00.000 ІЗ

Група ІІ-22-2

Дирда Степан

2026

Івано-Франківський національний технічний університет нафти і газу

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Дирда Степан Русланович

(прізвище, ім'я, по батькові)

УДК 004.942

(індекс)

БАКАЛАВРСЬКА РОБОТА

Програмна підтримка інтелектуальних систем прийняття рішень

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня С.Р. Дирда
(підпис, ініціали та прізвище здобувача)

Науковий керівник Пасека Надія Мирославівна, к.т.н., доцент
(підпис, прізвище, ім'я, по батькові, науковий ступінь, вчене звання керівника)

Допущено до захисту

Завідувач кафедри

доц. Бандура В.В.
(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківський національний технічний університет нафти і газу

Інститут, факультет інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти бакалавр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедрою

ІПЗ

доцент.

В.В. Бандура

“ ” 2026 р.

ЗАВДАННЯ НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ

Дирді Степану Руслановичу

(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи) "Програмна підтримка інтелектуальних систем прийняття рішень"

керівник проекту (роботи) Пасєка Надія Мирославівна, к.т.н., доцент

затверджені наказом вищого навчального закладу від “ 21 ” квітня 2026 р. № 179/7

2. Строк подання студентом проекту (роботи) 10 червня 2026 р.

3. Вихідні дані до проекту (роботи) Результати і матеріали отримані під час проходження переддипломної практики

4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)

1. Аналіз вимог до програмного забезпечення

2. Аналіз вимог і постановка задачі

3. Проектування системи

4. Реалізація проекту

5. Тестування та впровадження

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

1 Концептуальна архітектура i-DMSS (рис. 1.1, ст. 18)

2 Приклад функціонального відображення (рис. 2.1, ст. 33)

3 Цикл проектування (рис. 2.3, ст. 36)

4 Процес прийняття рішень (рис. 3.1, ст. 45)

5 Схема тришарової прямої нейронної мережі зі прихованим шаром (рис. 3.4, ст. 51)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

2. Дата видачі завдання 2026 р.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту	Примітка
1	Визначення та обґрунтування теми роботи	17.01.2026	виконано
2	Огляд існуючих концепцій, рішень та сервісів в даній області	21.02.2026	виконано
3	Побудова моделі або алгоритму власного рішення	01.03.2026	виконано
4	Документування реалізації власного оригінального рішення вибраними засобами	21.04.2026	виконано
5	Оформлення пояснювальної записки бакалаврської роботи	02.05.2026	виконано

Студент _____ Дирда С.Р.
(підпис)

Керівник роботи _____ Пасека Н.М.
(підпис)

АНОТАЦІЯ

Бакалаврська робота містить 63 сторінок, 14 рисунків, 9 таблиць, список 40 використаних джерел.

Метою роботи є дослідження методів і засобів програмної підтримки інтелектуальних систем прийняття рішень, а також аналіз підходів до побудови ефективних, адаптивних і масштабованих програмних рішень у цій галузі.

Об'єкт дослідження: інтелектуальні системи підтримки прийняття рішень.

Предмет дослідження: методи, моделі та програмні засоби реалізації інтелектуальних систем підтримки прийняття рішень, включаючи використання штучного інтелекту, нейронних мереж та еволюційних алгоритмів.

Результати дослідження: проведено аналіз сучасних підходів до побудови інтелектуальних СППР, досліджено архітектуру та принципи функціонування таких систем, розглянуто основні методи моделювання.

У першому розділі розглянуто теоретичні основи систем підтримки прийняття рішень, їх класифікацію, архітектуру та роль штучного інтелекту.

У другому розділі досліджено методи та інструменти розробки програмної підтримки СППР, зокрема підходи системної інженерії, життєвий цикл розробки та сучасні технології реалізації таких систем.

У третьому розділі розглянуто практичні аспекти реалізації інтелектуальних СППР, включаючи моделювання процесів прийняття рішень, застосування нейронних мереж та еволюційних алгоритмів.

Висновок: використання інтелектуальних методів у системах підтримки прийняття рішень дозволяє підвищити точність і швидкість прийняття рішень, забезпечити адаптивність систем та підвищити ефективність процесів.

КЛЮЧОВІ СЛОВА: СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ, ШТУЧНИЙ ІНТЕЛЕКТ, НЕЙРОННІ МЕРЕЖІ, ЕВОЛЮЦІЙНІ АЛГОРИТМИ, ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ, МОДЕЛЮВАННЯ, ПРОГРАМНА ПІДТРИМКА.

ANNOTATION

The bachelor's thesis contains 63 pages, 14 figures, 9 tables, and a reference list with 40 number of sources.

The aim of the work is to study methods and tools for software support of intelligent decision support systems, as well as to analyze approaches to building efficient, adaptive, and scalable software solutions in this field.

Research object: intelligent decision support systems.

Research subject: methods, models, and software tools for implementing intelligent decision support systems, including the use of artificial intelligence.

Research results: modern approaches to the development of intelligent decision support systems were analyzed, the architecture and operating principles of such systems were studied, key methods of decision-making process modeling were considered, and the application of artificial intelligence techniques to improve decision-making efficiency was examined.

The first section describes the theoretical foundations of decision support systems, their classification, architecture, and the role of artificial intelligence in their development.

The second section analyzes methods and tools for developing software support for decision support systems, including system engineering approaches, development lifecycle, and modern implementation technologies.

The third section covers practical aspects of implementing intelligent decision support systems, including modeling decision-making processes, the use of neural networks and evolutionary algorithms, and evaluation of system effectiveness.

Conclusion: the use of intelligent methods in decision support systems improves the accuracy and speed of decision-making, ensures system adaptability to environmental changes, and enhances the efficiency of management processes.

KEY WORDS: DECISION SUPPORT SYSTEMS, ARTIFICIAL INTELLIGENCE, NEURAL NETWORKS, EVOLUTIONARY ALGORITHMS, INTELLIGENT SYSTEMS, MODELING, SOFTWARE SUPPORT.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ ПРИЙНЯТТЯ РІШЕНЬ	9
1.1 Основні поняття, класифікація та еволюція систем підтримки прийняття рішень	9
1.2 Інтелектуальні системи підтримки прийняття рішень та їх архітектура	19
1.3 Роль штучного інтелекту у розвитку систем підтримки прийняття рішень	25
1.4 Висновки по розділу	30
РОЗДІЛ 2. МЕТОДИ ТА ІНСТРУМЕНТИ РОЗРОБКИ ПРОГРАМНОЇ ПІДТРИМКИ СППР	31
2.1 Методи проектування та життєвий цикл розробки інтелектуальних СППР	31
2.2 Інтеграція підходів системної інженерії та теорії прийняття рішень	35
2.3 Сучасні технології та інструменти реалізації інтелектуальних СППР	39
2.4 Висновки по розділу	44
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ ІНТЕЛЕКТУАЛЬНИХ СППР	45
3.1 Моделювання процесів прийняття рішень у програмних системах	45
3.2 Використання нейронних мереж у системах підтримки прийняття рішень	51
3.3 Застосування еволюційних алгоритмів для оптимізації рішень	55
3.4 Висновки по розділу	61
ВИСНОВКИ	62
СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА	64

					БР.ІП –05.00.00.000 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Програмна підтримка інтелектуальних систем прийняття рішень Пояснювальна записка	Літ.	Арк.	Акрушів
Розроб.		Дирда С.Р.					7	
Перевір.		Пасека Н.М.						
Реценз.		Храбатин Р.І.				ІФНТУНГ ІП-22-2		
Н. Контр.		Піх М.М.						
Затверд.		Бандура В. В.						

ВСТУП

У сучасному світі інформаційні технології відіграють ключову роль у забезпеченні ефективного управління складними системами та процесами. Зростання обсягів даних, підвищення рівня невизначеності та складності управлінських задач обумовлюють необхідність використання інтелектуальних систем підтримки прийняття рішень. Такі системи дозволяють автоматизувати процес аналізу інформації, підвищити обґрунтованість рішень та зменшити вплив людського фактору. Використання методів штучного інтелекту, зокрема нейронних мереж та еволюційних алгоритмів, відкриває нові можливості для створення адаптивних і ефективних програмних рішень у різних галузях діяльності.

Актуальність теми зумовлена необхідністю розробки сучасних програмних засобів, здатних забезпечити підтримку прийняття рішень в умовах невизначеності, багатокритеріальності та динамічності середовища. Існуючі системи підтримки прийняття рішень часто мають обмежену адаптивність, складність інтеграції з новими технологіями та недостатню ефективність при обробці великих обсягів даних. У зв'язку з цим виникає потреба у дослідженні та впровадженні інтелектуальних методів, які дозволяють підвищити продуктивність, точність і гнучкість таких систем.

Метою роботи є дослідження методів і засобів програмної підтримки інтелектуальних систем прийняття рішень, а також розробка підходів до побудови ефективних програмних рішень із використанням сучасних технологій штучного інтелекту.

Завданнями дослідження є аналіз предметної області та існуючих підходів до побудови систем підтримки прийняття рішень, визначення їх основних характеристик і архітектурних особливостей, дослідження методів та інструментів розробки програмної підтримки СППР, аналіз застосування нейронних мереж і еволюційних алгоритмів, моделювання процесів прийняття рішень, а також оцінка ефективності розроблених рішень.

Об'єктом дослідження є інтелектуальні системи підтримки прийняття

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

рішень як складні програмні системи, що забезпечують автоматизацію процесу аналізу даних і формування управлінських рішень.

Предметом дослідження є методи, моделі та програмні засоби реалізації інтелектуальних систем підтримки прийняття рішень, включаючи алгоритми штучного інтелекту, підходи до моделювання та сучасні технології розробки програмного забезпечення.

Методи дослідження включають аналіз наукових джерел для вивчення теоретичних основ, порівняльний аналіз існуючих підходів, методи математичного та імітаційного моделювання процесів прийняття рішень, експериментальні методи для реалізації та тестування програмних рішень, а також аналіз результатів для оцінки їх ефективності.

У роботі передбачається розгляд архітектури інтелектуальних СППР, дослідження сучасних методів їх реалізації та створення моделей, що забезпечують ефективну підтримку прийняття рішень. Особлива увага приділяється використанню нейронних мереж і еволюційних алгоритмів як основи інтелектуалізації таких систем, що дозволяє підвищити їх адаптивність і точність.

Бакалаврська робота містить 63 сторінок, 14 рисунків, 9 таблиць, 3 розділи, висновки та список 40 використаних джерел.

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ ПРИЙНЯТТЯ РІШЕНЬ

1.1 Основні поняття, класифікація та еволюція систем підтримки прийняття рішень

Особи, що приймають рішення, стикаються зі все більш стресовим середовищем – висококонкурентним, швидким, майже в режимі реального часу, перевантаженим інформацією, даними, розподіленими по всьому підприємству, та багатонаціональним за масштабом. Поєднання швидкості та доступу до Інтернету, а також розвитку методів штучного інтелекту, призвело до появи складних допоміжних засобів для підтримки прийняття рішень у цих ризикованих та невизначених умовах. Ці допоміжні засоби мають потенціал для покращення процесу прийняття рішень, пропонуючи рішення, кращі за ті, що приймаються лише людиною. Вони стають дедалі доступнішими в різних галузях, від медичної діагностики до управління дорожнім рухом та інженерних застосувань [1]. Ми назвали ці допоміжні засоби інтелектуальними системами підтримки прийняття рішень (i-DMSS) і починаємо цей розділ з дослідження інтелекту в його застосуванні до допоміжних засобів прийняття рішень. Ці системи можуть суттєво впливати як на процес прийняття рішень, так і на його результат. Таким чином, ми пропонуємо враховувати обидва фактори при їх оцінці.

У дослідженні традиційних систем підтримки прийняття рішень (СППР) виявлено, що більшість емпіричних досліджень зосереджувалися або на процесно-орієнтованих, або на результатно-орієнтованих заходах оцінки, але не на обох в одному дослідженні. Крім того, для оцінки найчастіше використовувалися кількісні показники результатів, такі як збільшення прибутку або зниження витрат. Коли використовувалися кілька показників, показники, орієнтовані на результат та процес, представлялися як окремі значення без інтегративної оцінки загальної цінності СППР. У випадку i-DMSS особливо

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

актуальним є врахування кількох критеріїв оцінки. Використання інтелектуальних методів впливає не лише на результат рішення, але й на процес прийняття рішень, що зростає зі зростанням використання веб-функцій СППР та СППР у режимі реального часу. Такі системи вдосконалюють та розширюють традиційні СППР, надаючи такі функції, як інформація «точно вчасно», обробка в режимі реального часу, обробка онлайн-транзакцій, підключення та глобально актуальна інформація.

Хоча традиційні DMSS досліджувалися ретельно, мало досліджень, якщо такі взагалі були, розглядали уніфіковану архітектуру для оцінки i-DMSS. Необхідність багатокритеріального оцінювання DMSS була відзначена в літературі. Наприклад, запропонували багатокритеріальний підхід, що враховує точки зору різних зацікавлених сторін. Запропоновано багатогранний підхід до оцінювання DMSS та експертних систем, що складається з суб'єктивних, технічних та емпіричних методів оцінювання [2]. Суб'єктивні методи були запропоновані для оцінки системи з точки зору користувачів та спонсорів; метод технічного оцінювання зосереджувався на аналітичних методах, що використовуються в DMSS; а емпіричні методи зосереджувалися на порівнянні продуктивності з системою та без неї. Запропонований підхід розширює попередні дослідження, пропонуючи уніфіковану архітектуру для оцінювання з використанням процесу аналітичної ієрархії (АНР), зосереджуючись при цьому саме на підтримці прийняття рішень.

Попередні дослідження показали, що багатокритеріальне оцінювання i-DMSS може бути проведене за допомогою АНР, враховуючи як результати, так і процес, щоб забезпечити єдину метрику. Однак дослідження конкретних категорій, які слід використовувати в рамках метрики АНР, не було визначено, і жодне дослідження не зосереджувалося виключно на i-DMSS. Отже, у цьому розділі оцінюється емпіричний i-DMSS з літератури з метою:

- (1) визначення критерію (або критеріїв), що використовуються для оцінки i-DMSS у літературі;
- (2) категоризація оціночних заходів;

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

- (3) пропонування відповідних категорій оцінювання для i-DMSS; та
- (4) рекомендація багатокритеріальної моделі для оцінки i-DMSS.

Слід зазначити, що неявним рішенням, яке має бути прийнято в моделі на основі АНР, є вибір найкращого дизайну кількох кандидатних архітектур i-DMSS на основі рангів, призначених внеском процесу та результату. Цей підхід є оцінкою *post mortem*, але з подальшими модифікаціями для прогнозування впливу його можна використовувати як метод апріорної оцінки. Отже, очікується, що концептуальний інструмент запропонованої оцінки запропонує керівництво для кращого архітектурного проектування та оцінки i-DMSS, враховуючи функції системи на основі штучного інтелекту в рамках оцінки процесу та результатів. Такий підхід пов'язує інвестиції у ІІІ, засновані на покращенні швидкості роботи комп'ютера та приросту пам'яті (фундаментальна перспектива інформатики), з вимірюваною індивідуальною, командною або організаційною продуктивністю.

У цьому розділі ми спочатку розглянемо загальний процес прийняття рішень та визначення інтелектуальних систем з літератури зі штучного інтелекту та DMSS, щоб встановити спільний словник та теоретичну базу [3]. Потім ми розглянемо компоненти загальних архітектур, запропонованих для i-DMSS, які включають рівні складності інформаційної системи. Далі пропонується інтегрована модель i-DMSS.

Нобелівський лауреат Герберт А. Саймон візуалізував процес прийняття рішень як пошук можливих шляхів у просторі пошуку. Ця візуалізація показує, що прийняття рішень також можна розглядати як « *процес вибору серед альтернативних напрямків дій з метою досягнення мети або цілей* ». У літературі описано кілька моделей того, як повинен проводитися або фактично проводиться процес прийняття рішень. Найпопулярніші базуються на моделі Саймона про три фази та нещодавні розширення. У таблиці 1.1 підсумовано розширену п'ятифазну модель.

Процес прийняття рішень є складним завданням. Він є безперервним та частково ітеративним, оскільки фази можуть перетинатися, і особа, яка приймає

рішення, може повернутися до попередньої фази. Однак, хоча деякі кроки можуть виконуватися одночасно, прийняття рішень є, по суті, послідовним процесом, де «проектування» вимагає «інтелекту», «вибір» потребує «проектування», а «впровадження» йде за «вибором». Ці кроки повторюються ітеративно з багатьма циклами зворотного зв'язку, доки остаточний вибір не буде реалізовано, а отримані уроки не будуть визначені та повідомлені.

Таблиця 1.1

Фази та кроки прийняття рішень

ФАЗА	КРОК	ОПИС
Інтелект (Intelligence)	Збір даних	Спостереження за реальністю та збір релевантних якісних і кількісних даних для загальної ситуації.
(Крок «формування порядку денного»)	Розпізнавання проблеми	На основі інтерпретації даних визначається чітке формулювання проблеми та загальна мета.
Проектування (Design)	Формулювання моделі	Використовуючи чітку проблему, створюється модель із набором варіантів дій, критеріїв результатів та параметрів.
(Крок «представлення проблеми»)	Аналіз моделі	Перевірка достовірності та пілотне тестування моделі для зменшення джерел значних помилок.
Вибір (Choice)	Генерація та оцінка	Оцінка всіх варіантів дій за допомогою аналізу «що-якщо», аналізу чутливості та пошуку цілей.
(Крок «пошуку та вибору альтернатив»)	Вибір	Пропонується найкращий варіант дій з використанням оптимізації або критеріїв задоволення.
Впровадження (Implementation)	Презентація результатів	Вибраний варіант подається керівництву для остаточного затвердження.
Навчання (Learning)	Аналіз процесу/результатів	Збір метрик процесу та результатів від команди, що приймала рішення.

Поняття інтелекту широко обговорюється в психологічній та суміжній літературі. Хоча стандартне визначення все ще залишається невловимим, існують деякі спільні характеристики «людського інтелекту», що ґрунтуються на літературі [4]. Такий інтелект включає здатність до навчання (*тобто* розширювати концептуальні та процедурні знання), розуміння та передачу повідомлень (*тобто* осмислювати повідомлення та генерувати очікувані відповіді), прийняття рішень та вирішення проблем раціональним способом, а

також розробку нових абстрактних та фізичних артефактів для виживання та розвитку в суспільстві.

У середині 1950-х років лауреат Нобелівської премії запропонував вивчати людську поведінку за допомогою інших методів, окрім стандартної та домінуючої парадигми психології. Саймон та його колеги прагнули розширити модель «порожнього організму», яка просто корелювала незалежні (*тобто* стимули) та залежні змінні (*тобто* реакції), вивчаючи пояснювальні механізми для організму. На відміну від тези Берінга, який стверджував, що пояснювальні механізми повинні бути дворівневими нейробіологічними, Саймон та його колеги висунули кілька шарів. Саймон зазначив, що *«теорії обробки інформації в когнітивних процесах представляють собою специфічний проміжний шар пояснення»*, встановили «гіпотезу про фізичну обробку символів». З цієї точки зору, людський розум є природною системою обробки символів, і дослідницькі зусилля ІІІ можуть бути зосереджені, головним чином, на розробці та тестуванні символічних систем з використанням комп'ютера як експериментального майданчика. Зокрема, припускається, що рівень інтелекту системи можна виміряти *«її здатністю досягати заявлених цілей в умовах варіацій, труднощів і складнощів, що виникають у середовищі завдання»*. Ці дослідники неявно припустили, що довгостроковою метою ІІІ є створення інтелектуальних артефактів, які імітують людських істот.

У статті під назвою «Обчислювальна техніка та інтелект» розробив тест для концепції. По суті, тест Тюрінга (ТТ) стверджував, що машину слід вважати розумною, якщо людина не може розрізнити взаємодію (через два комп'ютерні термінали) між комп'ютерною програмою та людиною, яка використовує один з терміналів. У цьому сенсі комп'ютерну програму можна вважати розумною, оскільки вона може імітувати раціональну та інтелектуальну людську поведінку [5]. У свою чергу, було представлено представив «евристичну гіпотезу». Згідно з цією гіпотезою, *«...всі нормальні люди мають однакові інтелектуальні механізми [тобто «запрограмованість»], і що відмінності в інтелекті пов'язані з кількісними біохімічними та фізіологічними умовами. Я розглядаю їх як*

швидкість, короткочасну пам'ять та здатність формувати точні та відновлювані довготривалі спогади». Також підтверджено існування загального спільного фактора інтелекту найвищого порядку, який називається g-фактором, для пояснення специфічних біохімічних та фізіологічних умов. Єнсен погоджується, що інтелектуальні дії (такі як « увага, сприйняття, дискримінація, узагальнення, навчання, пам'ять, мова, мислення, вирішення проблем тощо») є завданнями обробки інформації.

Використовуючи таку передумову, Маккарті припустив, що інтелектуальні артефакти базуються на інтелектуальних основних механізмах, запропонованих їхніми розробниками.

Ця ідея узгоджується з необхідністю вивчення шару пояснювальних механізмів для «порожнього організму».

Реальний потенціал побудови інтелектуальних систем також вивчався вченими, які не займаються штучним інтелектом. Зазначено, що ШІ має бути зосереджений на розробці нових алгоритмів, заснованих на надійних інженерних та наукових процесах. З цієї точки зору, емуляція евристики, подібної до людської, є складною через людські обмеження та неповноту методів спроб і помилок, що використовуються для розробки систем на основі правил [6]. Також захищали ту саму передумову: *«інформатика [стосується штучного інтелекту] - це неемпірична дисципліна. Ми б назвали її експериментальною наукою, але, як і астрономія, економіка та геологія, деякі з її унікальних форм спостереження та досвіду не відповідають вузькому стереотипу експериментального методу. Тим не менш, вони є експериментом»*. Системи посилення інтелекту «ІА > ШІ» кращі за системи штучного інтелекту. Ця точка зору підтримує поняття «слабкого ШІ», яке стверджує, що людський розум необхідно вивчати, щоб створювати кращі механізми для інтелектуальних систем, які допомагатимуть, але не замінюватимуть людей.

Поняття слабого ШІ підтверджує точку зору, що системи підтримки рішень повинні допомагати, але не замінювати людей у процесі прийняття рішень. Це поняття також узгоджується з передумовою Саймона про обмежену

					БР.ІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

раціональність у людей та концепцією створення кращих інструментів прийняття рішень для подолання та посилення людських можливостей.

Загалом, системи підтримки рішень (СППР) надають підтримку особам, що приймають рішення, об'єднуючи людські судження та комп'ютеризовану інформацію, намагаючись підвищити ефективність прийняття рішень. Загальну мету СППР можна сформулювати як «доповнення однієї або кількох здібностей особи, що приймає рішення». Попередники в розробці та дослідженні інтелектуальних СППР, які називалися системами підтримки рішень на основі знань (СКПР), запропонували такі характеристики для таких систем (друге покоління СППР, орієнтованих на те, щоб стати сучасними i-DMSS): (а) вони містять різні типи знань, що описують вибрані аспекти світу особи, що приймає рішення; (б) вони мають здатність отримувати та підтримувати описові знання, такі як ведення обліку та інші типи знань; (в) вони можуть створювати та представляти знання різними способами; (д) воно може вибирати знання для представлення або отримання нових знань; (е) воно може безпосередньо [інтелектуально] взаємодіяти з особою, яка приймає рішення.

Огляд основної літератури з DMSS у медичному, військовому, фінансовому, політичному та екологічному контекстах допомагає запропонувати об'єднану перспективу [7]. Для підтримки прийняття рішень необхідна інтелектуальна підтримка, подібна до людської, але остаточні та критичні рішення повинні приймати люди, які приймають рішення.

i-DMSS розширює традиційні DSS, включаючи методи для забезпечення інтелектуальної поведінки та використовуючи потужність сучасних комп'ютерів для підтримки та покращення процесу прийняття рішень. i-DMSS може, наприклад, швидко та успішно реагувати на нові дані та інформацію без втручання людини, вирішувати складні та запутані ситуації, навчатися на попередньому досвіді, застосовувати знання для розуміння середовища, розпізнавати відносну важливість різних елементів у рішенні, враховувати знання експертів у предметній області, рекомендувати дії та/або діяти від імені людини (за заздалегідь визначеним дозволом особи, яка приймає рішення).

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15

У деяких публікаціях у літературі підкреслюється внесок штучного інтелекту в дисципліну DSS [8]. Однак нещодавнє дослідження виявило кризову ситуацію в цій галузі. Незважаючи на чудові результати перевірки концепцій за допомогою штучного інтелекту для проектування i-DMSS, ці концепції ефективно підтвердили лише деякі кроки та фази прийняття рішень за останні 20 років. У вищезгаданому дослідженні було розроблено концептуальну архітектуру i-DMSS з метою оцінки, з точки зору верхнього рівня, основних рівнів можливостей, доступних у DMSS.

Таблиця 1.2.

Рівні можливостей інтерфейсу користувача (UI)

Рівень виміру UI	Опис можливостей
I	СППР (DMSS) надає мову дій у вигляді структурованих команд/меню та мову представлення на основі тексту та нединамічної графіки.
II	СППР надає мову дій (команди/меню) та мову представлення на основі гіпертексту, мультимедіа, звуків, відео або динамічної симуляції.
III	СППР надає мову дій у вигляді природної мови та мову представлення на основі середовищ віртуальної реальності.

Концептуальна структура оцінки можливостей для DMSS (CAF-DMSS) була оновлена з попередніх версій, опублікованих тими ж авторами. По суті, розширення включають новий вимір підтримки користувацького інтерфейсу та консолідацію попередніх рівнів обробки. Як результат, ця нова структура має три осі як основні структурні компоненти: (i) можливості користувацького інтерфейсу, (ii) представлення даних, інформації та знань, та (iii) механізми підтримки обробки.

Перший та другий виміри базуються на загальній та стандартній структурі для DMSS [9]. Третій вимір базується на типах завдань прийняття рішень, рівнях інтелекту, вбудованих в алгоритми, та типах інтелектуальних операцій для інтелектуальних систем аналізу даних, запропонованих у літературі. Попередня структура, про яку повідомляли автори, та ця інтегрують результати з літератури та встановлюють концептуальну порядкову шкалу для трьох вимірів. Перша вісь (можливості користувацького інтерфейсу) поділена на

три рівні, які досліджують багатство мови презентації та дій. У таблиці 1.2 наведено опис цих рівнів можливостей.

Вісь представлення даних, інформації та знань поділена на п'ять рівнів структурної складності, як описано в таблиці 1.3.

Таблиця 1.3.

Рівні можливостей представлення даних, інформації та знань (DI&K)

Рівень DI&K	Опис можливостей
I	Використання простих файлів, структур даних або одновимірних схем баз даних.
II	Використання складних структурованих даних або багатовимірних схем баз даних.
III	Доступ до структурованих даних та знань у кількісних моделях (прогнозування, нейронні мережі тощо).
IV	Доступ до слабоструктурованих знань: правила «якщо-то», семантичні мережі, фрейми, скрипти.
V	Доступ до мережі неструктурованих даних та баз знань на основі онтологій та розподілених систем.

Вісь обробки описує ступінь вбудованого інтелекту в досліджуваній DMSS, як описано в таблиці 1.4.

Таблиця 1.4.

Рівні можливостей обробки (P)

Рівень P	Опис можливостей
I	Всі SQL-подібні дії (пошук, оновлення) та OLAP-операції (зрізи, деталізація) для чіткої логіки.
II	Всі OLAP та SQL-подібні дії для нечітких (fuzzy) даних.
III	Сервіси класифікації, кластеризації, аналізу трендів для кількісних даних (генетичні алгоритми, data-mining).
IV	Алгоритми для складного аналізу якісних і кількісних даних: діагностика, моніторинг, інтерпретація.
V	Сервіси для синтезу: планування, дизайн, навчання. Гібридні інтелектуальні алгоритми.

Ці таблиці представляють порядкові концептуальні шкали для вимірювання ступеня: (i) можливостей користувацького інтерфейсу, (ii) необробленості компонента даних та (iii) ступеня

інтелект, вбудований в алгоритми або механізми обробки конкретної DMSS. Слід зазначити, що будь-який рівень підтримки включає або може включати можливості попередніх рівнів [10]. Отже, ми стверджуємо, що ця структура корисна для аналізу можливостей підтримки минулих, поточних та майбутніх DMSS, таких як i-DMSS.

Теоретична архітектура i-DMSS – Рисунок 1.1 – ілюструє приклад того, як ці можливості можна реалізувати. Обчислювальна потужність та пам'ять необхідні для доступу та аналізу величезних обсягів даних, а методи штучного інтелекту необхідні для розвитку інтелектуальної поведінки, необхідної для складних ситуацій прийняття рішень.

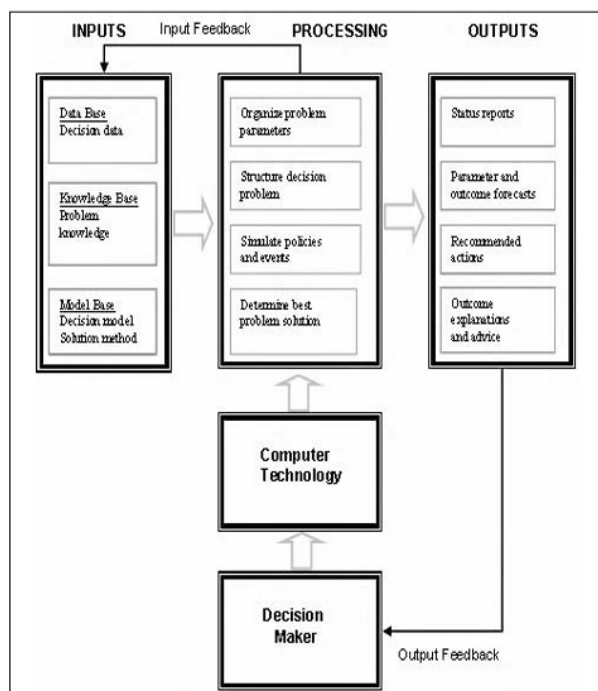


Рисунок 1.1 - Концептуальна архітектура i-DMSS

Отже, якщо залишити осторонь методологію, інтелектуальні системи мають характеристики, які зазвичай асоціюються з людським інтелектом. Відповідно, ми дотримуємося позиції, що інтелектуальна система повинна демонструвати людський інтелект, а також базовий механізм генерації інтелекту. Чим більше інтелектуальної поведінки демонструє система (тобто функціональності), тим вона інтелектуальніша, але ця поведінка має бути

спричинена використанням базових інтелектуальних механізмів (тобто структури) [11].

1.2 Інтелектуальні системи підтримки прийняття рішень та їх архітектура

Запропоновано та підтримали спостереження, що дослідження DSS (систем підтримки рішень) подолали дискусію про відмінності між DSS та ІІІ та тепер можуть зосередитися на тому, як кожен галузь можна використовувати в інтегрований дослідницькій програмі. З 1992 року підтримка цього спостереження просунулася від ранніх стадій інтеграції ІІІ в підтримку рішень, що характеризуються використанням експертних систем, нейронних мереж та інших основних методів ІІІ, до включення рівня абстракції підтримки рішень, який охоплює захоплення та застосування уподобань користувача як контексту для інтелектуального прийняття рішень [12]. Подальші рівні абстракції, що демонструють інтеграцію ІІІ та DSS в епоху знань, яка розглядає побудову моделей у системах підтримки рішень як процедуру, засновану на знаннях, та враховує компроміс між якістю моделі та її придатністю для використання в процесі прийняття рішень з обмеженим часом, які розглядають роль автономних агентів у DSS як одного з «персональних обчислювальних помічників», що забезпечують підтримку сумісності (інтеграція гетерогенних програм та мереж) та користувацьких інтерфейсів (персоналізована фільтрація інформації). Хоча змогли знайти достатню підтримку пропозиції, заснованої на знаннях, на індивідуальному рівні, дослідження в цій підгалузі за останнє десятиліття продемонстрували захопливий ступінь зростання від технічних коренів впровадження ІІІ в рамках індивідуальної підтримки рішень до вищих рівнів абстракції, які враховують вплив штучного інтелекту на налаштування, час та сумісність. З огляду журналу « Системи підтримки прийняття рішень » за останнє десятиліття можна знайти мало статей про системи групової підтримки, і ще менше - тих, що використовують штучний інтелект як довідкову

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

дисципліну. Винятки включають роботу, де спільне стратегічне планування покращується розподіленою агентною системою підтримки прийняття рішень, яка вбудовує стратегічні та предметні знання між самими агентами, які співпрацюють з користувачами за допомогою дощок та передачі повідомлень. Також розглядається теорія, що стосується ефективної співпраці автономних агентів у контексті підтримки прийняття рішень.

Можливо, відсутність доказів використання штучного інтелекту в дослідженнях систем підтримки груп пов'язана з колаборативними обчисленнями як дисципліною, що охоплює концепцію групової підтримки. Колаборативні обчислення історично спрямовані на вирішення питань комунікації за відсутності моделей прийняття рішень, як це можна знайти в дослідженнях систем підтримки прийняття рішень (СППР). У 1992 році було передбачено, що машинні інтелектуальні агенти взаємодіятимуть з користувачами-людьми через спільну платформу системи підтримки прийняття рішень, щоб залучити спеціалізовані знання, навички та досвід. Хоча ця передумова залишається актуальною, враховуючи кілька статей, знайдених у *системах підтримки прийняття рішень*, схоже, що штучний інтелект мав більший вплив на дослідження СППР на організаційному рівні.

Пропозиція, заснована на знаннях, передбачає впровадження знань у системи підтримки організаційних рішень (ODSS). Це впровадження, якщо воно буде процесорієнтованим за своєю природою, дозволить обійти традиційні бюрократичні процедури, що виникають як функція самої організаційної структури [13]. Як результат, ODSS має призвести до спрощення процесів та зниження операційних витрат. Однак на момент цієї публікації в організаційній підгалузі III/DSS було проведено мало роботи.

Більше десятиліття потому, організаційна підгалузь процвітала за схемою, подібною до тієї, що передбачалася в 1992 році. Низка робіт зосереджена на впровадженні знань в ODSS з точки зору захоплення знань, доступу до знань та метрик. Представлено цілеспрямовану схему моделювання для захоплення та організації знань у процесі прийняття рішень, яку можна

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

розширити шляхом збереження моделі рішень за допомогою вітрин моделей та сховищ. У таких роботах, досліджували використання отриманих знань для підтримки рішень за допомогою необмеженого системного мислення з метою переорієнтації ODSS на «погані» рішення. У відповідь на посилену увагу до ODSS на основі знань, пропонують у своїй архітектурі сховища знань, що ефективність DSS зрештою буде вимірюватися її здатністю просувати та розширювати знання, а також тим, наскільки добре вона покращує ментальні моделі та розуміння осіб, які приймають рішення, а отже, і їхнє прийняття рішень.

Дослідження ODSS, як і передбачалося в 1992 році, розглядало знання, орієнтовані на процес, як інструмент для спрощення процедур працівника знань. У роботі функція закупівель виграє від агентної ODSS, яка порівнює пропозиції постачальників у контексті знань щодо попередньої ефективності постачальників, щоб спростити рішення щодо компромісу між вартістю та якістю [14]. Стратегічне управління також виграє від конвергенції знань про процеси з бізнес-аналітикою, де стратегічні цілі вбудовані в алгоритми аналізу даних, які потім здатні адаптуватися та знаходити належні показники стратегічної ефективності. Подібні досягнення в підтримці прийняття рішень на основі знань, орієнтованих на процес, враховували як маркетинг, так і управління проектами.

Одним із наслідків пропозиції, заснованої на знаннях, у контексті ODSS є те, що організаційні структури можуть стати більш гнучкими та мати потенціал для зменшення формальності, оскільки складність процесу прийняття рішень всередині фірми перекладається на саму ODSS. Докази цього можна побачити як на тактичному, так і на стратегічному рівнях. Ціноутворення та призначення техніків, дві тактичні сфери, які зазвичай покладаються на ієрархічні структури управління для прийняття рішень, показані як хороші кандидати для прийняття рішень на основі штучного інтелекту на основі моделей прийняття рішень, заснованих на знаннях. Докази того, що ці дві сфери працюють так само добре або краще, ніж на платформі на основі штучного інтелекту/DSS, свідчать про те,

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

що історична структура управління для прийняття рішень у цих сферах більше не потрібна.

Інформація, необхідна для прийняття стратегічних рішень в організації, зазвичай вимагає взаємодії між відділами (*тобто* необхідна інформація є міжвідділовою), що часом може виявитися несвоєчасним та дорогим. У роботі пропонується багатоагентна система для моніторингу критичних факторів успіху, результати якої можуть бути корисними для процесу прийняття стратегічних рішень, водночас уникаючи постійної міжвідділової взаємодії для цілей моніторингу. У роботі використовується експертно-системний підхід для оптимізації SWOT-аналізу шляхом системної ідентифікації внутрішніх сильних та слабких сторін.

Ще одним наслідком тези, що базується на знаннях, є підвищення раціональності осіб, які приймають рішення на рівні організації. Це означає, що особи, які приймають рішення, можуть отримати користь від ODSS (системи, що базується на знаннях), в епоху знань, оскільки сама ODSS може враховувати більше інформації на більшій глибині, ніж особа, яка приймає рішення самостійно. Результатом є рішення, яке досліджує більший простір проблеми за менший час. Експертна система на основі знань обробляє зміни в плануванні, доступності та несправностях для гнучкої виробничої системи, де управління товарами переноситься на ODSS, яка одночасно застосовує кілька моделей прийняття рішень до проблеми рівнів товарів з плином часу.

Новішим розвитком у підтримці організаційних рішень, що доповнює прогнози 1992 року, є дослідження, спрямовані на взаємодію та координацію інструментів прийняття рішень і даних [15]. Пропонується розробка сховища метаданих для управління та підтримки деталей інтеграції. З цієї точки зору можна спостерігати, що дослідження в галузі інтеграції компонентів DSS відродять інтерес до гнучкості організаційної структури та зменшення складності міжвідомчої взаємодії. Ця складність походить від «культури обслуговування», яка зараз поширена у філософії управління сучасних організацій, включаючи ті підрозділи, яким зазвичай доручено відповідальність

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

за елементи проектів корпоративних обчислень (КОМ). Нові, дуже складні проекти КОМ вимагають попереднього етапу конфігурації для оцінки необхідного рівня залучення кожного підрозділу з метою створення надійного та ефективного рішення. Цей етап конфігурації призводить до «моделі залучення», яка включає деякі або всі ці сервісні підрозділи. Коли кожен сервісний підрозділ зосереджується на своїй дисциплінарній спеціалізації, отримана модель взаємодії вимагає значних витрат на інтеграцію через брак спільної лексики, альтернативних перспектив проблемної області, відсутності спільного розуміння інструментів кожної дисципліни, що використовуються в аналізі цієї проблемної області *тощо*. Крім того, персонал, який зазвичай самоучка, щоб стати ефективними міжсервісними інтеграторами підрозділів, часто буває настільки затребуваним, що він стає вузьким місцем для ефективного та економічно вигідного виконання портфеля поточних проектів. Тому, як тільки складність взаємодії системи стає значно нижчою, вузьким місцем для безперебійного налаштування та використання ресурсів буде організаційний, а не технічний характер.

З 1992 року бізнес-ландшафт змістився від акценту на вертикальній інтеграції до акценту на ключових компетенціях та віртуальній інтеграції між сукупністю фірм, що складають ланцюг або мережу поставок, що додає вартість. Розподілений характер ланцюгів поставок впливає на здатність будь-якої окремої фірми отримувати доступ до інформації поза межами власного локусу контролю. Це зумовлює потребу в підтримці рішень, яка виходить за межі традиційних меж фірми та охоплює ті фірми, які пов'язані та співпрацюють для виробництва товарів і послуг.

Розроблено структуру для агент-орієнтованих DSS як процес розробки скоординованого робочого процесу для співпрацюючих агентів, здатних підтримувати процес вирішення проблем у всій сукупності координуючих фірм. Ця структура доповнена пропозицією формальної мови запитів моделей та методів вибору та оптимізації моделі для міжорганізаційної координації на основі моделей та агентів.

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

Незалежність фірм у ланцюжку поставок створює додатковий вимір складності для міжорганізаційної підтримки рішень [16]. Деякі проблеми, що виникають з цього, включають неможливість інтеграції моделей та даних між фірмами, необхідність забезпечення рівнів абстракції моделі процесів для збереження прав власності окремих фірм та невідповідність стимулів для досягнення глобально оптимальних рішень (збільшення загальної вартості в ланцюжку поставок) порівняно з тими, які є локально оптимальними для кожної фірми (розширення частки окремої фірми у вартості ланцюжка поставок).

Питання інтеграції моделі та даних розглядалося як з точки зору ієрархічної композиції даних з використанням підходу простору фільтрів, так і з точки зору метаданих, який описує контент, релевантний для DSS, як внутрішній, так і зовнішній по відношенню до організації. Однак управління як просторами фільтрів, так і метаданими є унікальним для кожної окремої конфігурації ланцюга поставок і вимагає переробки у разі зміни мережі фірм-учасниць. Це обмежує швидкість, з якою ланцюги поставок можуть переналаштовуватися, і створює бар'єр для автоматизації ланцюгів поставок у сенсі інтелектуальної автоматизованої реконфігурації для адаптації до збурень у мережі в режимі реального часу.

Контроль гранулярності моделі є вимогою в міжорганізаційних системах підтримки рішень у будь-якому випадку, коли процес прийняття рішень в одній фірмі вимагає інформації про модель в іншій фірмі, яка пов'язана з детальним та власним процесом та/або моделлю прийняття рішень. Використання станів діяльності/процесу з використанням представлень процесу як механізм абстрагування внутрішніх процесів від інтерфейсу, представленого зовнішнім фірмам у міжорганізаційній системі підтримки рішень.

Невідповідність стимулів у ланцюжку поставок стосовно підтримки прийняття рішень, здається, є відкритим питанням, яке може бути корисним з економічної літератури з розробки стимулів [17]. Ранні роботи в цій галузі пропонують формальну мову специфікації контрактів на основі XML, яку можна використовувати для забезпечення стимулювання шляхом укладання контрактів

на рівні самої системи підтримки прийняття рішень. Цей метод є привабливим завдяки своїй реалізації на системному рівні, що надає можливість дослідженням ШІ/СППР розглянути наслідки автоматизованих переговорів щодо контрактів для участі в процесах прийняття рішень у ланцюжку поставок. Потенціал для вторинного ринку даних і моделей у ланцюжку поставок і вплив цього ринку на традиційні ролі в ланцюжку поставок є цікавою екстраполяцією динаміки міжорганізаційної мережі підтримки прийняття рішень.

1.3 Роль штучного інтелекту у розвитку систем підтримки прийняття рішень

Виникнення штучного інтелекту як еталонної дисципліни для DSS досліджувалося раніше. Однак, замість «виникнення», зв'язок між цими двома галузями краще описати як спільне походження. Адже ці дві дисципліни не лише виникли приблизно в один і той самий час (кінець 1950-х років), але й мають спільний початок. Відомий як піонер штучного інтелекту та засновник науки про прийняття рішень, якою ми її знаємо сьогодні. Його книга «Нова наука про управлінські рішення», яка об'єднує ці два основні напрямки його мислення та досліджень, забезпечує хороший контекст для спільного розуміння поглядів Саймона та історії цих двох галузей.

У першому розділі своєї книги, присвяченому впливу комп'ютерних технологій на процеси управління, Саймон виділяє три виміри розбіжностей серед експертів щодо ступеня цього впливу: технологічний вимір, філософський вимір та соціально-економічний вимір. Грубо кажучи, за словами Саймона, думки експертів різняться залежно від того, наскільки малим чи великим вони вважають вплив. Узгоджуючи перші два виміри, Саймон визнає чотири можливі школи думки в цьому відношенні (див. Таблицю 1.1) та характеризує власну позицію як «досить екстремальну за всіма вимірами» – а саме як технологічного радикала, економічного консерватора та філософського прагматика. Відповідно до цієї позиції він потім робить такі твердження та прогнози щодо ролі

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

комп'ютерів в організаціях:

Я вірю, що в наш час комп'ютери зможуть виконувати будь-яке когнітивне завдання, яке може виконати людина. Я вірю, що комп'ютери вже можуть читати, думати, навчатися, творити [18]. Я вірю, що комп'ютери та автоматизація сприятимуть постійному, але не значному прискореному, зростанню продуктивності, що повна зайнятість буде підтримуватися попри це зростання, і що людство не знайде життя виробництва та споживання в більш автоматизованому світі дуже відмінним від того, яким воно було в минулому.

Майже через сорок років після цих пророцтв справді відрезвляє уявлення про те, скільки з них ще не здійснилися (*наприклад*, повна зайнятість та зникнення бідності та злиднів, які він передбачає пізніше у своїй книзі), а скільки інших досі обговорюються. Однак особливе значення для цієї дискусії має радикальний погляд Саймона на можливості комп'ютерів, особливо стосовно їхнього стосування в організаційних та управлінських процесах. Щоб зрозуміти джерело цього радикалізму, нам потрібно вивчити погляди Саймона на людське пізнання та прийняття рішень у організаціях.

Історію розвитку штучного інтелекту можна умовно поділити на чотири періоди, в яких домінують чотири основні підходи, які я називатиму: i) класичний підхід, ii) підхід, заснований на знаннях, iii) конекціоністський підхід та iii) ситуативний підхід. Хоча між ними є певні збіги, кожен з них відрізняється основними рисами та передумовами (див. таблицю 1.5).

Класичний ІІІ розглядав пізнання як абстрактне (фізичне втілення не має значення), індивідуальне (одинокий розум є основним осередком інтелекту), раціональне (міркування є парадигматичним для інтелекту) та відокремлене (мислення відокремлене від сприйняття та дії). Спираючись на ці принципи, ранній ІІІ, як показано на прикладі PSSH, справді досяг великих успіхів. Системи, побудовані на цих принципах, були відносно успішними у вирішенні абстрактних проблем, але вони раптово зазнали невдачі у вирішенні більш конкретних завдань та областей, які, здавалося б, здавалися буденними та тривіальними для людей, *наприклад*, розпізнавання літер алфавіту, переклад між

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

мовами, навігація в неідеалізованих місцевостях тощо.

Таблиця 1.5

Основні підходи до ШІ та їх ключова ідея

Підхід	Основна ідея
Класичний	Маніпулювання символами
На основі знань	Знання
На основі прецедентів	Спогади / Аналогії
Конекціоністський	Розподілені обчислення
Ситуативний	Втілення та вбудованість (Embodiment)

Невдача класичного підходу у вирішенні цих проблем та виконанні своїх обіцянок призвела до зниження інтересу до досліджень штучного інтелекту з боку фінансових установ на початку 1980-х років, що спонукало практиків шукати практичні проблеми для вирішення.

Оскільки знання стали вважати ключем до таких практично-орієнтованих зусиль, на сцені з'явився новий клас артефактів («експертні системи») та нова група практиків («інженери знань»). Завдання інженера знань розглядалося як тричастинне:

- Витягти з експерта, *тобто* «людини, чия голова містить знання», знання (які вони інтуїтивно розуміють), які вони використовують у своїй галузі знань;
- Формалізувати ці знання, використовуючи відповідну мову представлення, у базі знань;
- Порівняти ефективність цієї формалізації з ефективністю експерта з метою «виправлення» поведінки машини, *тобто* наблизити її до інтроспективного опису експерта.

Практика інженерів знань та ранній успіх експертних систем принесли штучному інтелекту повагу, якої він прагнув, не лише в академічних колах, а й у бізнесі, де, за деякими даними, мільярди доларів було інвестовано в експертні системи для виробництва, фінансових послуг, діагностики машин тощо. Але цей успіх був дуже обмеженим через крихкість цих систем. Незважаючи на очевидну витонченість та експертизу в спеціалізованих галузях, таких як медицина,

експертні системи продемонстрували явну нерозуміння дуже базових фактів, які людина сприймає як належне.

Спроби виправити цю ситуацію досі здебільшого не увінчалися успіхом, що найкраще видно на прикладі проекту. Творці вважали, що для досягнення інтелекту людського рівня нам потрібно забезпечити машину достатньою кількістю знань, щоб вона могла самостійно продовжувати процес їх отримання [19]. Тому вони взялися за створення величезної бази знань, яка після певного моменту мала б навчатися самостійно та виходити «за межі людського знання». Це виявилось недосяжною метою, зокрема, через величезний, неявний та контекстуальний характер великої кількості людських знань. Організації стикаються з подібними проблемами у роботі зі знаннями.

Підхід, заснований на знаннях, супроводжував два інші погляди. Один з них – це міркування, засноване на прецедентах, яке наголошує на спогадах, а не на знаннях. Інший – це так званий погляд на планування, який розглядає людську діяльність як фундаментально сплановану та добре продуману. Однак, як і підхід, заснований на знаннях, обидва ці погляди зіткнулися з нездоланими перешкодами.

Конекціоністський підхід (знову) виник у 1980-х роках і захопив уяву багатьох дослідників, зокрема деяких у DSS. Головною особливістю цього підходу була його опозиція до явних форм знання та акцент на архітектурах, подібних до мозку, але він залишався відданим більшості принципів класичного ІІІ [20]. Зокрема, він зберіг поняття ментальної репрезентації, яке було центральним у класичних теоріях розуму. Тому, незважаючи на початковий запал, конекціонізм не зміг відвернути ІІІ від деяких його фундаментальних передумов. Це було залишено на розгляд пізніших розробок, особливо тих, що пов'язані з ситуативним поглядом на пізнання.

Ситуативний погляд змінює багато припущень попередніх підходів, особливо класичного ІІІ. Тобто, він розглядає інтелект як втілений (важливе фізичне втілення), вбудований (важливе безпосереднє природне та соціальне середовище), орієнтований на дію та значною мірою імпровізаційний.

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

Ці концептуальні зрушення в підходах до ШІ (які ми обговоримо детальніше пізніше) мають значення не лише для нашого розуміння людського інтелекту, але й, як я стверджую, для проектування таких артефактів, як комп'ютеризовані системи підтримки прийняття рішень. Щоб мати змогу сформулювати ці наслідки, нам також потрібно простежити за розвитком СППР з моменту їх одночасного виникнення зі ШІ.

Штучний інтелект та СПР зазнали різючих змін з моменту їхнього спільного виникнення близько півстоліття тому. У процесі цього розвитку ці дві галузі взаємно інформували одна одну, але більшість їхніх дій відбувалася на технічному рівні. Спільнота СПР, зокрема, завжди була «в курсі» останніх технічних розробок у сфері ШІ див.табл. 1.6 та інших комп'ютерних технологій. Однак, як стверджує Чіборра у наведеній вище цитаті, це не завжди призводило до найефективніших, змістовних, корисних та зручних для користувача систем. Цей розділ є спробою виправити цю ситуацію шляхом узгодження СПР з останніми розробками у сфері ШІ. Дотримуючись підходу «зверху вниз», ми почали з концептуального рівня та вивели основні принципи проектування, які, у свою чергу, привели нас до методологічного (і частково технічного) рівнів. Результатом цього дослідження є перспектива проектування iDMSS з такими основними моментами:

Таблиця 1.6

Розвиток СППР у зв'язку з комп'ютерними технологіями

Етап	Період	Домінуюча концепція	Технології
I	1960-ті – 1970-ті	Моделювання даних та вирішення проблем	Бази даних, MIS
II	1980-ті	Групова підтримка рішень (GSS)	Бази знань, експертні системи, EIS
III	1990-ті	Організаційне навчання та управління знаннями	OLAP, сховища даних, data mining
IV	2000-ні	Веб-орієнтовані та активні СППР	Інтернет, клієнт-сервер, програмні агенти

- Вводячи діяльність у картину, ця перспектива наголошує на процесі прийняття рішень, а не на їхньому результаті;

- Применшуючи значення ментальних моделей, це зменшує когнітивне навантаження на тих, хто приймає рішення, пов'язане з обмірковуванням;
- Починаючи із зовнішньої ситуації, це збільшує ймовірність і, ймовірно, легше для кількох осіб, які приймають рішення, дійти спільного уявлення про проблему (що, можливо, є важливим кроком до досягнення консенсусу);
- Завдяки включенню імпровізації збільшується ймовірність більш рефлексивних та адаптивних результатів.

Це ключові особливості концептуального та методологічного підходу, представленого тут. Підсумовуючи, головний урок останніх розробок у галузі штучного інтелекту для проектування i-DMSS полягає в тому, що значна частина людського пізнання відбувається не в індивідуальних, відокремлених та безтілесних мізках, а в контекстуальних взаємодіях між втіленими людьми – або, метафорично кажучи, пізнання відбувається в дикій природі. Це урок, який спільнота DSS повинна сприйняти найсерйозніше [21]. Від зали засідань до аудиторії та від штаб-квартири до польових робіт – це має бути девізом майбутніх розробників i-DMSS.

1.4 Висновки по розділу

У першому розділі було розглянуто теоретичні основи інтелектуальних систем підтримки прийняття рішень. Проаналізовано основні поняття, класифікацію та етапи еволюції систем підтримки прийняття рішень, що дозволило визначити їх місце у сучасних інформаційних технологіях. Досліджено архітектуру інтелектуальних СППР та ключові компоненти їх побудови. Також розглянуто роль штучного інтелекту у розвитку таких систем, зокрема у підвищенні точності та ефективності прийняття рішень. Отримані результати формують теоретичну основу для подальшого дослідження методів та програмної реалізації СППР.

РОЗДІЛ 2. МЕТОДИ ТА ІНСТРУМЕНТИ РОЗРОБКИ ПРОГРАМНОЇ ПІДТРИМКИ СППР

2.1 Методи проєктування та життєвий цикл розробки інтелектуальних СППР

Процеси проєктування та розробки СППР були предметом кількох порівняльних досліджень у минулому. Кожне з цих досліджень класифікує процеси за різними критеріями. Наприклад досліджується та аналізується десять методологій СППР за парадигмою, структурою та орієнтацією. Для цілей свого огляду парадигма стосується моделей, що лежать в основі методологій (орієнтовані на рішення, керовані на процес, керовані на дані або системні). Структура вказує на підхід, який використовується для керівництва процесом розробки (етапний або ситуативний). Орієнтація включає в себе рекомендації щодо розвитку, прийняті дослідниками СППР (нормативні або описові). Огляд Арінзе призводить до моделі ситуацій для вибору методології СППР.

Інший приклад таких порівняльних досліджень, який аналізує дванадцять методологій DSS та випадків використання розробки DSS з особливим акцентом на еволюції DSS. Він пропонує структуру, засновану на етіології, походженні та темпі еволюції [22]. Для цілей його структури етіологія стосується причин еволюції (екзогенні чи ендогенні тригери), «походження [стосується] того, чи відбувається еволюція в межах програми чи між програмами, а темп стосується схеми еволюції з часом» (безперервна еволюція, пунктурна рівновага або квантова еволюція). Структура уточнює природу еволюції DSS та «може допомогти системним аналітикам передбачити, що може статися далі в процесах розробки, та допомогти їм вирішити, які методи та інструменти, ймовірно, будуть успішними з кожним класом еволюції».

Для цілей нашого дослідження ми пропонуємо різний огляд та аналіз дев'яти процесів розробки СППР з урахуванням їхньої основної спрямованості. Деякі процеси чітко зосереджені на «прийнятті рішень», інші – на «системній

інженерії», а треті – на обох. Розуміння відмінностей між цими трьома категоріями потім використовується для пропозиції інтегративної, тристоронньої структури розробки. У таблиці 2.1 коротко підсумовано розглянуті методології.

Таблиця 2.1.

Розглянуті методології розробки систем підтримки прийняття рішень

Методологія	Фокус	Особливості
Функціональне відображення	Прийняття рішень	Співвідношення функцій СППР з організаційними підрозділами.
Граф рішень	Прийняття рішень	Визначення рішень на оперативному, тактичному та стратегічному рівнях.
Прототипування	Системна інженерія	Створення часткових версій програми для розробки фінального продукту.
Дизайн-цикл	Інтегрований фокус	Еволюційний дизайн, що пов'язує підтримку рішень із реалізацією.

Однак дослідження з розробки СППР не завершилися в 1991 році. Дослідники та практики продовжують регулярно пропонувати нові або модифіковані рішення. Однак більшість цих нових методологій залишаються сильно натхненними роботами попереднього покоління, і ми вирішили представити в цьому огляді рішення, які отримали широке визнання спільноти.

Процеси розробки СППР, зосереджені на прийнятті рішень, були особливо популярними на початку досліджень СППР. Ми вибрали три процеси, представлені їхніх відповідних авторів між 1979 і 1983 роками, і які досі регулярно цитуються в сучасній літературі з DSS [23]. Процеси представлені в хронологічному порядку.

На рисунку 2.1 відтворено приклад функціонального відображення з для СППР, побудованої для визначення фінансового та логістичного впливу запропонованих змін у продуктивній лінійці виробничої корпорації.

Автор стверджує, що методологія функціонального картування може допомогти скоротити час розробки проектів СППР «шляхом визначення існуючих модулів, які можна модифікувати для включення до системи». Картування також може допомогти координувати проект «шляхом чіткого

визначення відповідальності за модифікацію модулів», а також може допомогти зменшити фрагментацію зусиль інформаційних систем шляхом «визначення існуючих джерел інформації, які можуть сприяти розвитку СППР».

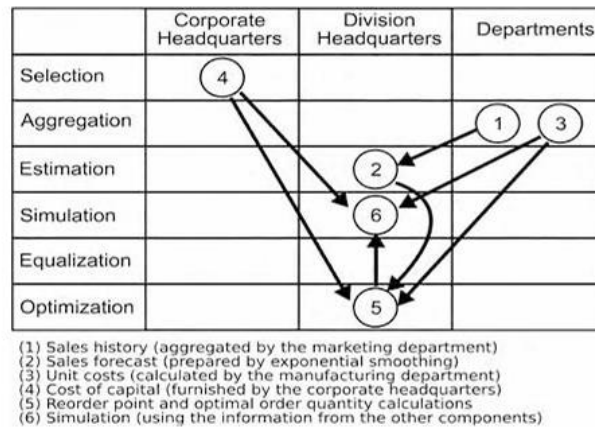


Рисунок 2.1 - Приклад функціонального відображення

Методологію *графів рішень* було запропоновано як модифікацію описової моделі системної динаміки. Ця методологія «підкреслює графічні, а не комп'ютерні результати моделювання; змінює термінологію з інженерної на орієнтовану на рішення контекст; та дозволяє використовувати стандартний комп'ютерний шаблон».

Методологія є суто графічною та використовує символи, натхненні структурами системної динаміки (рис. 2.2).

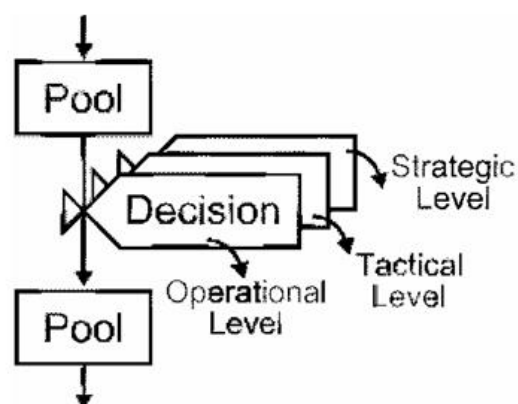


Рисунок 2.2. Швидкість руху ресурсів між резервами визначається рішенням

Графіки рішень використовуються для створення моделі рішень з метою визначення відповідних рішень. «Ці рішення поділяються на підрішення на операційному, тактичному та стратегічному рівнях управління. Після того, як ці підрішення виділено, елементи даних ідентифікуються шляхом спостереження, інтерв'ю, дедукції або комбінації цих трьох». У цій моделі пули – це накопичення ресурсів, а рішення визначають швидкість, з якою ресурси перетікають з одного пулу до іншого.

Процес розробки СППР, орієнтований на прийняття рішень, у відповідь на техноцентричні, системно-орієнтовані методології розробки, запропоновані на початку 1980-х років. На його думку, саме «рішення в концепції повинно визначати унікальний контекст СППР, і це рішення повинно мати наслідки для того, чому, як і що потрібно будувати під час побудови таких систем». В іншому випадку обмежена роль «Р» у СППР призводить до незрозумілих систем, які часто мають непередбачувані наслідки. На думку автора, важливе розмежування між суттю (що використовується під час прийняття рішень) та процедурою (як приймаються рішення), оскільки зусилля з розробки легко зосереджуються виключно на сутності ситуації прийняття рішення.

Щоб приділити більше уваги процедурі, процес розробки спирається на три взаємопов'язані дії, які він разом називає дослідженням рішень:

- Збір даних, включаючи дані про поточне прийняття рішень з використанням різних методів (інтерв'ю, спостереження, анкетування та історичні записи);
- Описове моделювання, що створює узгоджений опис поточного процесу прийняття рішень;
- Нормативне моделювання, що визначає норму того, як слід приймати рішення.

Інші фази проектування та розробки СППР (такі як функціональні специфікації, впровадження, моніторинг та оцінка) слідує за дослідженням рішень. У цій орієнтованій на рішення нормативній методології акцент робиться на зміні існуючого процесу прийняття рішень для підвищення ефективності

прийняття рішень. На думку автора, «розробку СППР слід розглядати як спробу зосередити більше уваги на аспектах прийняття рішень менеджерами. Таким чином, літера «D» у СППР використовується в прескриптивному сенсі»

2.2 Інтеграція підходів системної інженерії та теорії прийняття рішень

У попередніх двох розділах було розглянуто деякі процеси розробки, що зосереджені або на «прийнятті рішень», або на «системній інженерії». У той час як процеси, що зосереджені на аспектах прийняття рішень, стосуються організаційних питань та надають рекомендації щодо проектування бази знань СППР, процеси, що зосереджені на системній інженерії, повинні надавати рекомендації щодо впровадження та підтримки цієї бази знань. У цьому розділі розглядаються процеси проектування, які намагаються інтегрувати обидва напрямки в одну об'єднуючу методологію. Центральним для більшості цих процесів є заявлена необхідність прийняття еволюційного, проміжного або адаптивного проектування, постійно рухаючись туди-сюди між потребами користувачів у підтримці рішень та способами їх впровадження в цілісну систему. Ми вибрали три методології розробки, запропоновані їхніми відповідними авторами між 1978 і 1991 роками. Методології представлені в хронологічному порядку.

Цикл проектування, запропонований можна вважати предком більшості інших процесів розробки СППР. Він широко цитується в літературі з СППР і досі вважається авторитетною моделлю. Глобальний цикл показано на рисунку 2.3. Для цілей цього огляду додано блоки, що позначають кроки підтримки прийняття рішень та кроки системної інженерії.

Тісний зв'язок, необхідному між функціональними можливостями підтримки рішень та їхньою реалізацією в реальній системі. Вони критикують «тенденцію команди розробників зосереджуватися на системі як на артефакті та втрачати з поля зору те, чого вона повинна досягати». На їхню думку,

недостатньо визначити СППР «з точки зору вхідних та вихідних даних» та «вказати конкретні дані та звіти». Іншими словами, «важливо наголосити на використанні СППР та її роботі, а не на її технічних характеристиках».

Їхній погляд на еволюційний дизайн повністю відповідає методології прототипування, представлений у попередньому розділі. Система, яка є зручною та корисною, повинна бути розроблена та впроваджена якомога швидше, але вона також повинна бути достатньо гнучкою, щоб дозволяти швидке розширення та додавання підпрограм [23].

Цикл проектування також включає аспекти, типові для системної та програмної інженерії. Наприклад, відокремлення інтерфейсу від функціональності системи (що називається «імперативами» на рисунку 2.3) являє собою традиційний шаблон проектування, що полегшує додавання підпрограм та модифікацію або розширення існуючих з часом.

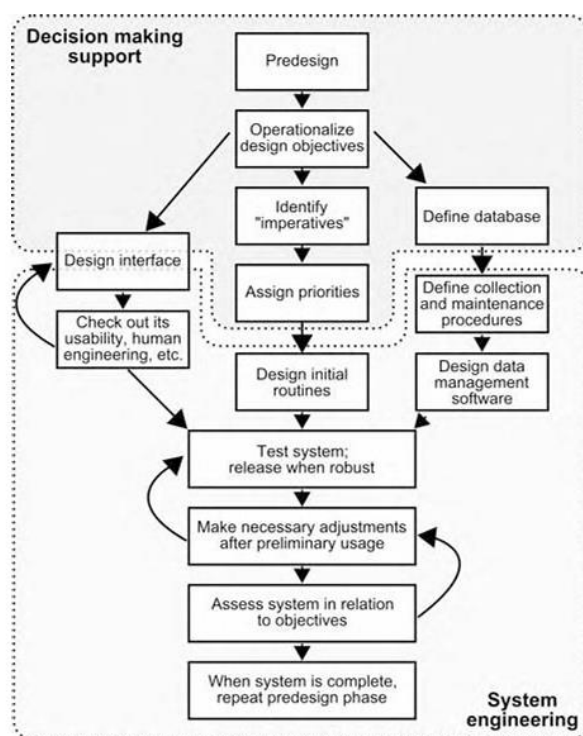


Рисунок 2.3 - Цикл проектування

Одним із недоліків цієї методології є те, що вона робить частину процесу, пов'язану з системною інженерією, лінійно залежною від частини проектування

підтримки рішень. Як ми побачимо в розділі, що представляє нашу інноваційну структуру, ця особливість має суттєві наслідки для розробки та підтримки фактичної СППР.

Спрейг та Карлсон (1982) пропонують методологію розробки СППР, популярну в літературі з СППР, яку можна розбити на дві широкі частини: «план дій», що складається з чотирьох фаз, описаних у таблиці 2.2, та методологію ROMC, «процесно-незалежну модель для організації та проведення системного аналізу в СППР». Підхід ROMC складається з чотирьох компонентів: представлень, операцій, допоміжних засобів запам'ятовування та механізмів керування (звідси чотирилітерна аббревіатура) [24]. У таблиці 2.3 описано ці компоненти з деякими прикладами.

Таблиця 2.2

План дій DSS

Фаза	Опис
I. Попереднє дослідження та оцінка доцільності	Опитування бази користувачів та оцінка їхніх потреб. Проведення пілотних проектів для визначення загальних характеристик та вимог до СППР.
II. Розробка середовища СППР	Формування групи розробки СППР, визначення її місії та зв'язків з іншими підрозділами організації. Встановлення мінімального набору інструментів і даних та введення їх в експлуатацію.
III. Розробка початкової специфічної СППР	Ідентифікація, аналіз та проектування першої конкретної СППР разом із користувачами. Оновлення інструментів і даних відповідно до потреб, що виникають у процесі роботи.
IV. Розробка наступних специфічних СППР	Оцінка потреб для подальших спеціалізованих систем. Розробка нових СППР на основі початкової моделі.

Хоча методологія ROMC більше схиляється до аспектів системної інженерії (хоча вона не уточнює, як кожен із чотирьох компонентів фактично реалізується), фази плану дій не можна розбити на фази, що зосереджені на прийнятті рішень, та фази, що зосереджені на системній інженерії, оскільки кожна фаза приділяє увагу обом видам операцій. Це можна пояснити кардинальним правилом, яке автори застосовують до методології: СППР має

керуватися потребами користувачів. Іншими словами, «зусилля щодо СППР не слід вживати, якщо потреба не є очевидною [25]. Однак оцінка ступеня потреби вимагає певного розуміння та зобов'язань щодо СППР заздалегідь (ще одне питання курки та яйця)».

Тому автори включають діяльність зі створення прототипів вже на першому етапі (у формі пілотних проектів) та використовують безперервну та ітеративну оцінку так званого «мінімального набору» інструментів та даних для поступового розширення проекту, доки він не стане конкретною СППР. Тим не менш, ця методологія залишається високорівневою за своєю природою та лише обмежено підтримує етап впровадження самої СППР.

Остання методологія цього огляду, «Інженерія підтримки прийняття рішень» (ІППР), запропонована як *«комплексна методологія, заснована на моделі життєвого циклу розробки СППР, яка охоплює інженерний підхід до аналізу та проектування СППР»*. Прототипування також є важливою частиною цієї методології. Процес проілюстровано на рисунку 2.4. Для цілей цього огляду додано блоки, що позначають кроки підтримки прийняття рішень та кроки системної інженерії. Кроки інженерії вимог та проектування СППР перетинаються в обох категоріях, оскільки деякі з їхніх підетапів належать до підтримки прийняття рішень, а інші – до системної інженерії.

Таблиця 2.3

Компоненти ROMC

Компонент	Опис	Приклади
Представлення (Representations)	Концептуалізації, що використовуються як методи комунікації між користувачем та СППР.	Діаграми, таблиці, звіти, форми введення даних, рівняння.
Операції (Operations)	Дії, необхідні для роботи СППР або для створення та доставки представлень.	Діагностика та структурування проблеми; збір, маніпулювання та перевірка даних; оцінка ризиків альтернатив; моделювання результатів.
Засоби пам'яті (Memory aids)	Компоненти, призначені для підтримки використання різних представлень та операцій.	Бази даних, робочі простори (workspaces), посилання, тригери для сповіщення користувачів, профілі користувачів.

Як показано на рисунку 2.4, методологія інженерії підтримки прийняття рішень складається з п'яти фаз, кожна з яких розбивається на низку завдань. Намагаючись подолати обмеження традиційних моделей життєвих циклів для розробки DSS, методологія DSE пропонує окремий етап для розуміння структури завдання перед спробою його моделювання. Подібно до циклу проектування, цей підхід є типовим для послідовного процесу, де завдання програмної інженерії йдуть за завданнями підтримки прийняття рішень. Автор широко використовує петлі зворотного зв'язку між різними фазами методології. Однак він не описує, як розробник може повернутися до попереднього етапу у разі невдачі мети на пізнішому етапі. Це проблема, що зустрічається в багатьох методологіях розробки DSS: *« багато статей просто стверджують, що був використаний еволюційний підхід без подальших уточнень »*.

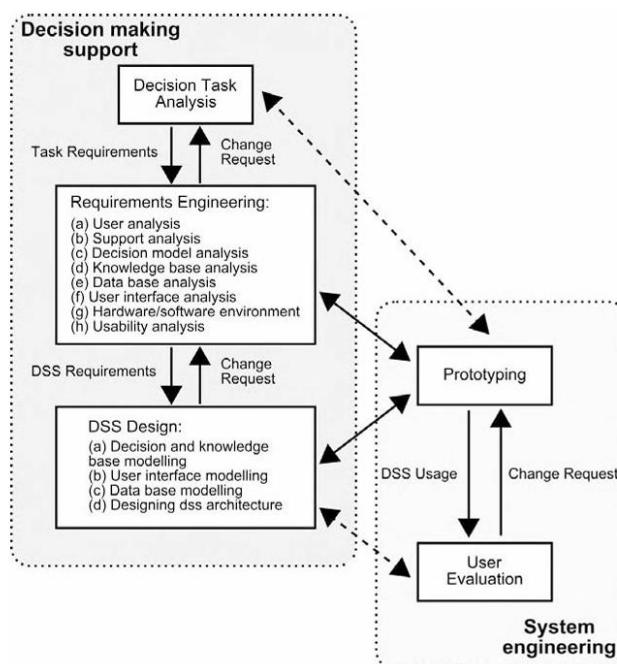


Рисунок 2.4 - Методологія інженерії підтримки прийняття рішень

2.3 Сучасні технології та інструменти реалізації інтелектуальних СППР

У цьому розділі повторно розглядаються дві інтегровані методології,

представлені в огляді, відповідно до структури, представленої в попередньому розділі. Йдеться про процес інженерії підтримки прийняття рішень та цикл проектування.

Ми обрали методологію інженерії підтримки прийняття рішень як перший приклад, оскільки її реорганізація відповідно до тристороннього підходу є дуже очевидною та ілюстративною [26]. На рисунку 2.5 показано переглянуту методологію інженерії підтримки прийняття рішень, і її слід порівняти з початковим процесом, зображеним на рисунку 2.4.

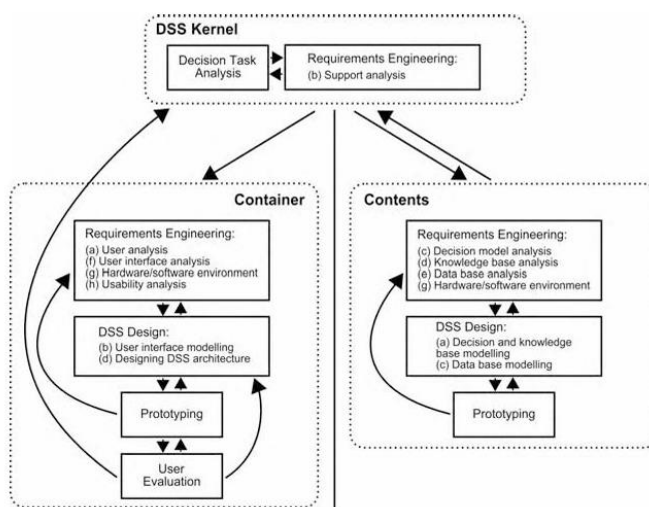


Рисунок 2.5 - Перегляд концепції «Інженерія підтримки прийняття рішень» відповідно до трикомпонентного підходу

Три компоненти фреймворку розробки (контейнер, вміст і ядро) чітко визначені на рисунку 2.5. Як і в оригінальній методології, процес починається з аналізу завдання прийняття рішень. Наступний крок оригінальної методології, інженерія вимог, розбивається на три основні компоненти. Літери, що використовуються в дужках для позначення підзадач, збережені, як і в оригінальному процесі (рисунок 2.4).

Аналіз підтримки, метою якого є отримання знань про очікування користувачів та «надавання набору цілей СППР, які визначають її підтримку», чітко визначає операції, які виконуватимуться СППР і які впливатимуть на розробку як контейнера, так і вмісту. З одного боку, вміст відповідає за надання

даних і моделей, необхідних для операцій. Цю відповідальність виконують аналіз моделі рішень, аналіз бази знань, аналіз бази даних та апаратно-програмне середовище, пов'язане з підзадачами СУБД та MBMS кроку розробки вимог. З іншого боку, контейнер відповідає за забезпечення базової системної інфраструктури та інтерфейсу СППР. Він пересилає вхідні дані користувача до механізму знань та відображає звіти та результати, що поширюються у відповідь. До цієї частини належать аналіз користувача, аналіз інтерфейсу користувача, аналіз зручності використання та апаратно-програмне середовище, пов'язане з інтерфейсом користувача та системною частиною СППР.

Так само, етап проектування DSS також можна розділити на контейнер та його вміст. Моделювання інтерфейсу користувача та проектування загальної архітектури системи належать до контейнера. Моделювання бази рішень та знань, а також моделювання бази даних, належать до частини, що відповідає за вміст [27]. Потім кожен частину можна реалізувати незалежно та ітеративно з використанням прототипів. Розбіжності, виявлені під час оцінки користувачем стосовно цілей, можуть призвести до коригування аналізу завдань та підтримки (у механізмі знань), що реалізується як модифікації як у контейнері, так і у вмісті. Зауважте, що вміст зазвичай оцінюється користувачем за допомогою всієї системи (разом з контейнером), а не окремо.

Важлива відмінність між оригінальним та переглянутим циклами проектування полягає в тому, що переглянутий цикл явно враховує незалежні цикли проектування та прототипування як для контейнера, так і для вмісту. Як пояснювалося раніше, кожен компонент потребує власного життєвого циклу розробки, оскільки він помітно відрізняється від інших. Як наслідок, кроки розробки, зосереджені на прийнятті рішень, та ті, що зосереджені на системній інженерії, не є послідовними на глобальному рівні (як на рисунку 2.4), а лише всередині окремих компонентів контейнера та вмісту [28]. Звідси природно випливає, що паралельне проектування та реалізація, а також повторне використання, значно покращуються, оскільки контейнер та вміст не повинні безпосередньо залежати один від одного. Такий підхід дозволяє досягти кращого

ітеративного проектування, ніж оригінальний монолітний процес.

Другою переглянутою методологією є цикл проектування. Ми обрали цей процес, оскільки він широко визнаний у літературі з DSS і досі впливає на багатьох дослідників і практиків. Цей приклад допомагає підкреслити переваги, які тристоронній підхід надає порівняно з існуючими процесами. Рисунок 2.6 ілюструє переглянутий цикл проектування і його слід порівняти з початковим циклом проектування, показаним на рисунку 2.6.

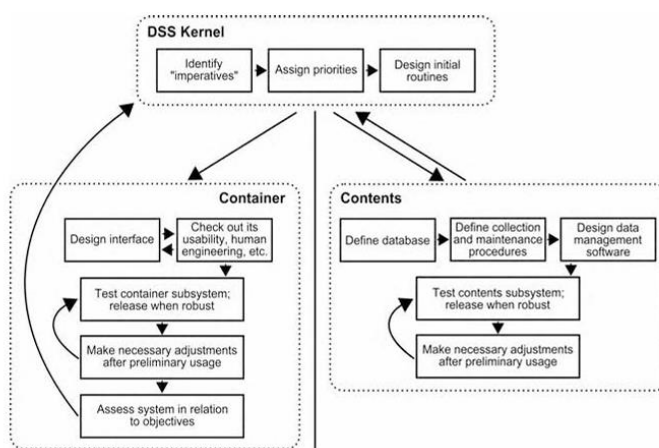


Рисунок 2.6 - Перегляд циклу проектування відповідно до тристороннього підходу

Як і в початковому циклі проектування, процес починається з визначення функціональних вимог до СППР («імперативів» або операцій, які повинна забезпечувати СППР). Фактична реалізація цих функціональних можливостей впливає на два інші компоненти підходу. З одного боку, контент відповідає за надання даних, необхідних для операцій (мали справу лише з даними у своєму початковому циклі проектування, але моделі, правила або документи можна було легко додати до частини контенту) та відповідних систем управління базами даних [29]. З іншого боку, контейнер відповідає за забезпечення базової системної інфраструктури та інтерфейсу СППР.

Розбіжності, виявлені під час оцінки системи стосовно цілей, можуть призвести до коригування імперативів, визначених механізмом знань, що буде реалізовано як модифікації як у контейнері, так і у вмісті.

Як і в переглянutoму процесі прийняття рішень, важлива відмінність між оригінальним та переглянутим циклами проектування полягає в тому, що переглянутий цикл явно враховує незалежні цикли тестування та коригування як для контейнера, так і для вмісту. Як наслідок, кроки розробки, зосереджені на прийнятті рішень, та ті, що зосереджені на системній інженерії, не є послідовними на глобальному рівні (як на рисунку 2.3), а лише всередині окремих компонентів контейнера та вмісту. Ті ж переваги з точки зору ітеративної розробки та повторного використання також застосовуються до цього переглянутого процесу розробки.

Справедливо буде сказати, що досягнення в управлінні знаннями та штучному інтелекті оновили сферу підтримки рішень, привнісши нову форму інтелекту в СППР. Однак ці досягнення також змінюють перспективу розробки СППР, яка раніше здебільшого стосувалась організаційних та технічних питань, і висвітлюють труднощі в адаптації до сучасних проблем управління знаннями.

На основі ретельного аналізу існуючих процесів та методологій розробки СППР у цьому розділі пояснюється, що процеси розробки суто технічних компонентів системи та бази знань СППР дуже відрізняються з точки зору часу розробки, парадигм, інструментів, технічного розвитку та експертизи, необхідної розробнику. Отже, пропонується нова основа для проектування та розробки СППР, наріжним каменем якої є чіткий розмежування між контейнером СППР (відповідальним за системну частину) та вмістом СППР (відповідальним за базу знань). На найвищому рівні інтерфейсний шар між контейнером та вмістом реалізується в компоненті, який називається ядром СППР. Цей компонент включає як базову модель підтримки рішень, так і механізм знань СППР. Ядро керує завданнями, що впливають на розробку як контейнера, так і вмісту СППР.

Цей інноваційний підхід все ще має кілька обмежень, які слід врахувати в майбутньому. Зокрема, він покладає значну відповідальність на компонент ядра, який повинен мати можливість взаємодіяти як з контейнером, так і з вмістом СППР. Ще одним обмеженням є те, що запропонований підхід не

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

враховує контекст цільової СППР, хоча досвід показує, що розробка великої стратегічної СППР для багатонаціональної компанії дуже відрізняється від розробки невеликої логістичної СППР для місцевої компанії. Іншими словами, жодну методологію розробки СППР не можна вважати універсальним рішенням, і наш підхід не є винятком.

Незважаючи на ці обмеження, які заслуговують на особливу увагу для майбутніх досліджень, запропонований фреймворк має важливі застосування, оскільки він може допомогти розробникам СППР краще зрозуміти окремі процеси розробки контейнера та вмісту СППР, а також зв'язки між двома компонентами через ядро СППР. Як тристороння надбудова, в якій можна повторно використовувати існуючі методології, що зосереджені на підтримці прийняття рішень або на розробці систем, фреймворк, представлений у цьому розділі, має допомогти галузі розробки СППР вийти з етапу розширення та перейти на етап управління. Нарешті, ми сподіваємося, що цей фреймворк надасть дослідникам та практикам інструмент для розробки кращих систем та архітектур СППР [30].

2.4 Висновки по розділу

У другому розділі було досліджено методи та інструменти розробки програмної підтримки інтелектуальних систем прийняття рішень. Розглянуто підходи до проєктування та життєвий цикл розробки інтелектуальних СППР, що забезпечують системність і ефективність створення таких систем. Проаналізовано інтеграцію методів системної інженерії та теорії прийняття рішень для побудови більш точних і адаптивних моделей. Також досліджено сучасні технології та інструменти реалізації інтелектуальних СППР, які забезпечують їх функціональність і масштабованість. У результаті визначено ключові підходи до створення ефективної програмної підтримки Інтелектуальних систем.

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ ІНТЕЛЕКТУАЛЬНИХ СППР

3.1 Моделювання процесів прийняття рішень у програмних системах

Процес підтримки рішень Загалом кажучи, рішення – це вибір, який полягає у досягненні певної мети шляхом аналізу суб'єктивно-об'єктивних умов, генерування альтернатив та вибору найбільш підходящої серед них. Загальний процес підтримки рішень можна описати як такий, що має такі інтерактивні аспекти: інтелект, проектування, вибір та реалізація, як показано на рисунку 3.1.

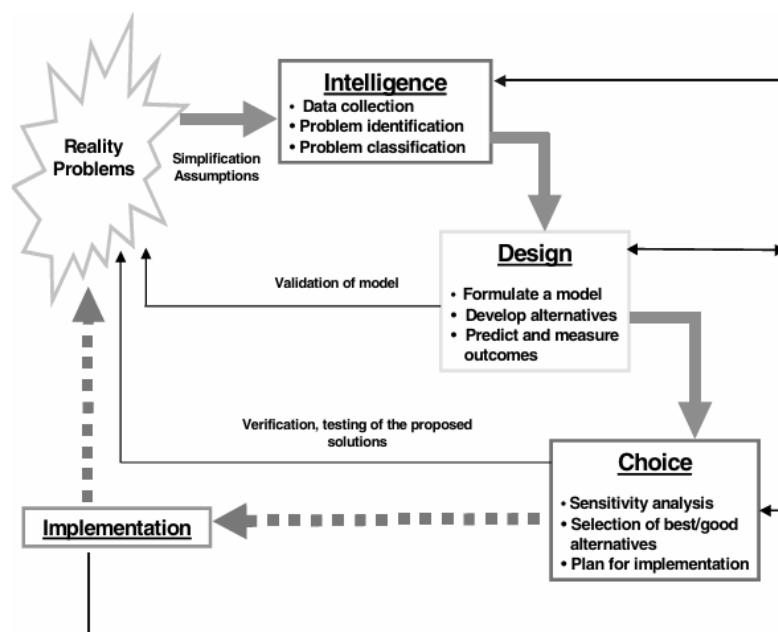


Рисунок 3.1 - Процес прийняття рішень

Він проходить етапи, починаючи від ідентифікації та класифікації проблеми, спрощення припущень, збору даних, формулювання моделі, створення альтернативних рішень, оцінки та вибору, а також валідації та верифікації моделі та тестування запропонованого рішення до остаточної реалізації плану [31]. Поточне дослідження зосереджено на тому, як підтримка знань може допомогти особі, яка приймає рішення, прийняти рішення під час процесу проектування. На рисунку 3.2 ілюструється сценарій впровадження

підтримки рішень на основі знань (DDS) з точки зору управління знаннями про рішення (DKM), в якому технології управління знаннями включають генерування та отримання знань, кодифікацію знань, а також обробку та використання знань (міркування) тощо .

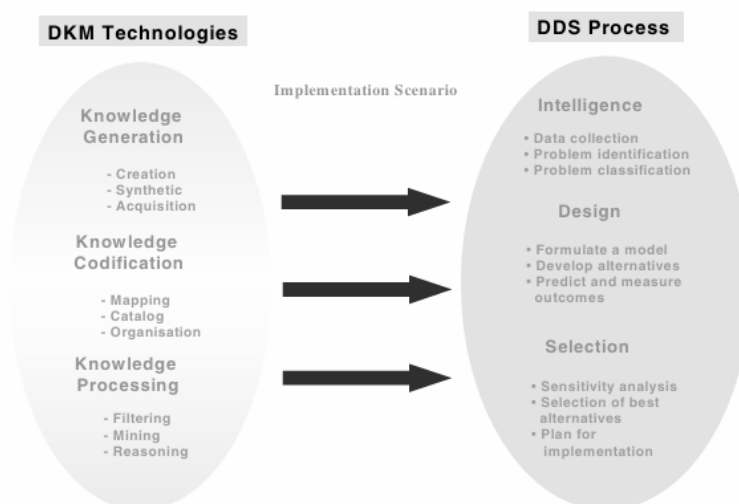


Рисунок 3.2 - Сценарій впровадження системи підтримки прийняття рішень

Головна роль дизайнера полягає в застосуванні наукових та інженерних знань для пошуку (генерації, оцінки та вибору) рішень проектних задач, а потім оптимізації цих рішень у рамках вимог та обмежень, встановлених фізичними, екологічними та людськими факторами. Ми розглядаємо дизайн як процес перетворення інформації, що характеризує потреби та вимоги до продукту, у знання про продукт. Виходячи з принципу проектування на основі рішень, рівняння дизайну можна виразити наступним чином: $\{ \mathbf{K} \} = \mathbf{T} \{ \mathbf{I} \}$, де, $\mathbf{K}$ – це вихід знань, $\mathbf{I}$ – вхідна інформація, а $\mathbf{T}$ відповідно, це взаємозв'язок трансформації. Таким чином, наукоємна підтримка стає більш критичною в процесі проектування та була визнана ключовою технологією, що сприяє збереженню конкурентної переваги в розробці продукту.

У цьому розділі ми представляємо розробку наукоємної схеми підтримки прийняття рішень щодо проектування, як зображено на рисунку 3.3, в якій підтримка прийняття рішень щодо проектування використовується на основі синтезу моделювання процесу проектування (DPM), управління знаннями (KM)

та підтримки прийняття рішень (DS). З мотивації та огляду процесу підтримки прийняття рішень щодо проектування видно, що теорії прийняття рішень, такі як теорія ігор, теорія корисності, теорія ймовірностей, теорія нечітких множин та теорія розширених множин *тощо* , відіграють ключову роль у цьому процесі.

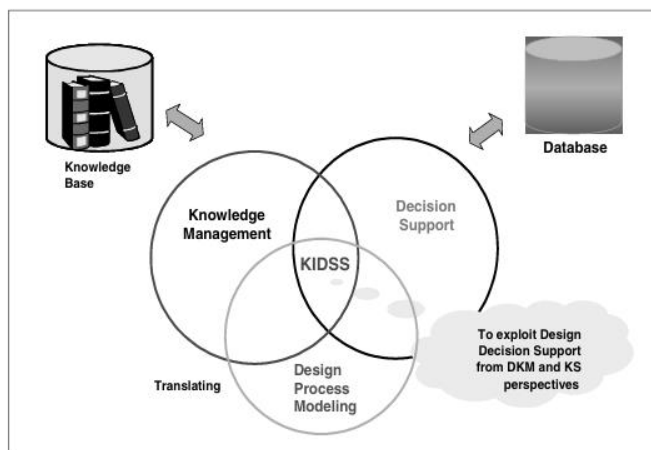


Рисунок 3.3 - Система підтримки проектування на основі знань (KIDSS)

Координація є центральною проблемою багатоагентних систем (MAS), яка включає співпрацю та проблему вирішення конфліктів. Фактично, проблема конфлікту є некооперативною проблемою, тому вирішення конфліктів є ключовою технологією для MAS. Поширеними способами вирішення конфліктів є арбітраж та переговори. Арбітраж базується на класичній математичній теорії та правилах міркування відповідно до концепції характеристичної функції, що означає «бути чи не бути, (так, ні) або $(0,1)$ », тоді як спосіб узгодження базується на теорії нечітких множин та правилах міркування відповідно до концепції функції належності, що означає, що ступінь належності «бути чи не бути, [так, ні] або $[0,1]$ ». Обидва способи призводять до втрати інтересів окремих агентів різного ступеня. У цьому розділі два механізми, трансформуючий міст та регуляторний перемикач, використовуються для вирішення конфліктних або несумісних проблем та співпраці/переговорів між дизайнерами/особами, що приймають рішення в дизайні.

Метод трансформуючого мосту пропонується для вирішення конфліктів або проблем несумісних рішень. Двом сперечаючимся сторонам потрібен міст,

щоб перекинути річку, яка їх розділяє. Річка є нейтральною територією, і тому міст повинен фізично трансформуватися, щоб служити місцем для вирішення конфлікту. Після трансформації «кімната» більше не повинна функціонувати як міст. Типовим уявним прикладом використання цього методу є те, як люди в Гонконзі вирішують так звану проблему «зв'язків між правим та лівим рухом». Таким чином, концепція трансформуючого мосту дозволяє двом несумісним сторонам перемогти, зберегвши свої власні два специфічні інтереси. Це не метод вирішення, заснований на конкуренції, і не той, у якому компроміс та баланс не є їхніми інтересами.

Ми розглядаємо проблеми конфлікту як протилежні (сумісні та несумісні) проблеми. Трансформація базується на правилах міркування відповідно до концепції залежної (передаточної) функції, що означає, що «бути чи не бути можна взаємозамінити». На основі принципів «йти в протилежному напрямку» та «диспозиції частоти кількох гілок» можна розробити трансформуючий міст для вирішення проблем конфлікту ресурсів між багатьма агентами та проблем конфлікту динамічного проектування. Зокрема, трансформуючий міст проектування, який може грати пов'язану та трансформовану роль, проектується між двома гравцями таким чином, щоб конфлікт можна було вирішити, і таким чином два гравці могли отримати задовільні рішення. Обробляючи несумісні проблеми, діапазон пошуку буде розширюватися крок за кроком відповідно до ступеня конфлікту проблем, що потребують вирішення. Наприклад, процес гри в шахи можна проаналізувати за допомогою цього мосту. Коли в небезпеці більше однієї шахової фігури, за допомогою звичайних оптимальних алгоритмів буде розрахована найменша втрата, а фігура з найменшою втратою буде відкинута. За допомогою стратегії розширення, вже розрахована найменша втрата буде використана для розширення набору та пошуку нового шансу серед тих фігур, які на даний момент не перебувають у небезпеці, але мають можливість розпочати нову атаку. Ціна цієї нової атаки має бути навіть меншою, ніж розрахована раніше втрата.

Традиційно, дизайнер зазвичай покладається на знання людини та метод

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

спроб і помилок при визначенні значення параметра. Однак ці методи нелегко застосовувати, коли системних параметрів з потенційними зв'язками забагато. Генетичний алгоритм має перевагу в пошуку оптимуму та уникненні локальних значень. Разом з генетичним алгоритмом розроблено метод систематичного налаштування параметрів для особи, яка приймає рішення в складному системному проектуванні. Представлення дизайну sGA використовує регуляторні гени, які діють як перемикач для ввімкнення (активний) та вимкнення (пасивний) генів. Кожен ген на вищих рівнях діє як перемикальний вказівник, який має дві можливі цілі: коли ген активний (увімкнено), він вказує на ціль нижчого рівня (ген), а коли пасивний (вимкнено), він вказує на ціль того ж рівня. На етапі оцінки лише експресовані гени особини транскрибуються у фенотипову функціональність, що означає, що лише гени, які є активними на даний момент, сприяють придатності рішення. Пасивні гени не впливають на придатність і переносяться як надлишковий генетичний матеріал під час еволюційного процесу прийняття рішень. Отже, використання підходу sGA до спільного проектування рішень можна підсумувати наступним чином. По-перше, гени представляють модулі або підсистеми рішень, які є активними або пасивними, залежно від того, чи сприяють вони вирішенню проблеми рішення. Потім, сімейство рішень рішень, що базуються на додаванні або відніманні модулів рішень, може бути оцінено шляхом чергування різних «активних» та «пасивних» модулів або підсистем. Таким чином, сімейство рішень відповідатиме варіантам моделі рішень, які мають різні активні та пасивні комбінації модулів або підсистем.

Під час прийняття рішень між кількома дизайнерами вкрай важливо узгодити кілька атрибутів дизайнерської угоди, таких як матеріал, метод виробництва, значення параметрів, вартість, кількість, якість та відносні переваги. Переговори – це форма прийняття рішень за участю двох або більше активно залучених агентів, які не можуть приймати рішення самостійно і тому повинні йти на поступки для досягнення компромісу [32]. Тому переговори щодо величезного обсягу транзакцій в Інтернеті стали фундаментальним механізмом

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

автоматизації спільного проектування. Крім того, гнучкість та адаптивність механізму переговорів можуть бути використані як правдоподібне джерело мотивації та основи для проектування інтелектуальних та автономних агентних систем. У цій роботі механізм переговорів з використанням моделі FSD складається з наступних шести фаз:

(1) Механізм переговорів запускається з «початкових пропозицій щодо дизайнерської угоди» агента (проектувальника). На цьому етапі кожен агент переговорів пропонує свої умови переговорів, що відображають його відносну перевагу щодо угоди. Дизайн-угода складається з кількісних умов, таких як значення параметрів та вартість. Однак нечіткі значення для цих умов змінюються нечіткими функціями належності, що відображають якісні умови, такі як відносна перевага.

(2) Після початкових пропозицій агента, для підтримки побудови нечіткої матриці попарних порівнянь A використовуються «нечіткі функції належності». Використовуючи цю нечітку функцію належності, відносні уподобання дизайнера перетворюються на нечіткі значення належності. Під час процесу перетворення можна використовувати нечіткі функції належності дзвоноподібної форми (або Z , X , n та S -типу).

(3) Будується «матриця попарних порівнянь A ». На цьому етапі матриця порівняння АНР використовується для обчислення відносної важливості кожної альтернативи (угоди). В результаті пропозиції кожного агента повністю порівнюються.

(4) Ця фаза призначена для «вибору бажаної пропозиції» переговорного агента. На основі результатів порівняння на фазі (3) один або кілька дизайнерів обирають бажані пропозиції. Однак це перший крок динамічного процесу переговорів.

(5) П'ятий етап призначений для «перегляду пропозицій та переговорів». На цьому етапі кожен агент переглядає свою «початкову пропозицію» та продовжує переговори зі своїм контрагентом. Для цього використовується методологія «пошуку мети» для перегляду початкових пропозицій.

(6) Заключний етап полягає у запропонованні «оптимальної пропозиції». Для запропоновання оптимальної пропозиції використовуються нечітка матриця попарних порівнянь А та механізм виводу АНР, а потім перехід до етапу (5) для досягнення консенсусу зі своїм колегою. В результаті кожен розробник може бути задоволений остаточними пропозиціями.

3.2 Використання нейронних мереж у системах підтримки прийняття рішень

Штучні нейронні мережі (або просто нейронні мережі, НМ) – це клас нелінійних регресійних моделей, дискримінантних моделей та моделей обробки даних, які тісно пов'язані між собою та працюють узгоджено для вирішення проблеми нейронні мережі (НМ) натхненні тим, як людський мозок обробляє інформацію. Ці корисні інструменти забезпечують механізм для аналізу великих обсягів даних та навчання на основі даних для пошуку закономірностей та виявлення нелінійних зв'язків. Засновані на навчанні, а не на попередньо запрограмованій поведінці, НМ принципово відрізняються від послідовного, логічно заснованого, запрограмованого підходу. НМ мають здатність узагальнювати на основі попередніх випадків, подібно до того, як люди використовують свої емпіричні спостереження та досвід, і, таким чином, мають здатність пропонувати рішення з неточних та складних даних. Застосунки, що використовують цю особливість виведення значення попередньої поведінки або моделей використовуються для надання рекомендацій щодо прийняття рішень, які можуть виходити за рамки алгоритмічних підходів [33].

Основним обчислювальним елементом є нейрон, який отримує вхідні дані та генерує вихідні дані за допомогою нелінійного перетворення зваженої суми своїх вхідних даних. Один нейрон має типову математичну форму

$$y = f(\sum_{i=1}^n w_i x_i) \quad (3.1)$$

де $x_1, x_2, \dots, x_n$ є входами нейрона, тоді як $w_1, w_2, \dots, w_n$ – це пов'язані зв'язки (ваги) для моделювання синаптичного навчання. Ваги можуть бути позитивними або негативними, тобто збуджуючими або гальмівними²³.

Архітектурно, нейронна мережа складається з колекції нейронів, розташованих шарами, які з'єднані мережею налаштованих числових з'єднань. Дві загальні топології нейронних мереж – це мережі прямого зв'язку та рекурентні (зі зворотним зв'язком) мережі. У нейронній мережі прямого зв'язку сигнали проходять від входів вперед через, можливо, багатошарові (які називають прихованими шарами) до виходів. Нейронні мережі прямого зв'язку виявилися найбільш корисними для задач прийняття рішень, оскільки вони протікають у спосіб, що узгоджується з процесом прийняття рішень. Рисунок 3.4 показано ілюстрацію 3-шарової мережі прямого зв'язку з прихованим шаром.

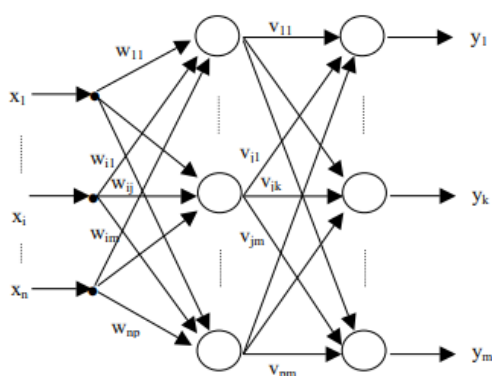


Рисунок 3.4 - Схема тришарової прямої нейронної мережі зі прихованим шаром

Рекурентні нейронні мережі дозволяють петлі зворотного зв'язку, як показано на рисунку 3.5, і можуть мати або не мати прихованих шарів. Вони можуть включати повний або частковий зворотний зв'язок залежно від моделі.

Сильною стороною нейронної мережі для задач прийняття рішень є її здатність наближати будь-яку обмежену неперервну функцію до будь-якої як завгодно малої похибки апроксимації. НМ «вивчає» функцію, використовуючи три основні стратегії: навчання з учителем, навчання без учителя та навчання з підкріпленням. У навчанні без учителя НМ не надається жодних виходів, пов'язаних з відповідними вхідними даними. Мета полягає у виявленні базової

структури даних, такої як кореляції або асоціації між шаблонами в даних. У навчанні з учителем НМ отримує навчальний набір даних, що складається з пар вхідних даних та відповідних виходів; НМ намагається знайти та скоригувати ваги на вхідних даних так, щоб виходи, отримані НМ, були якомога ближчими до вибірових виходів (або цілей).

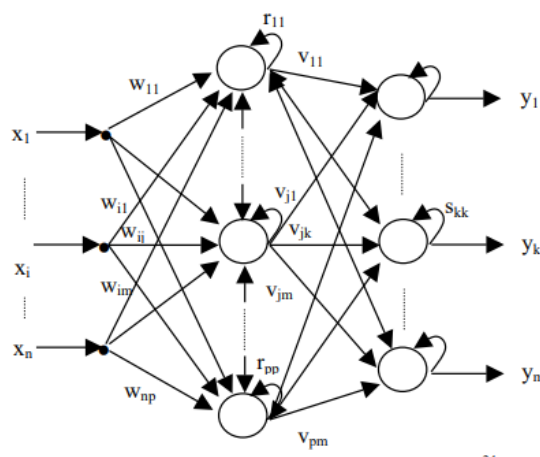


Рисунок 3.5 - Ілюстрація рекурентної нейронної мережі

Побудовану мережу потім можна використовувати для прогнозування виходу з новим набором вхідних даних, щоб забезпечити керівництво щодо прийняття рішень. У навчанні з підкріпленням НМ отримує лише узагальнені або високорівневі вказівки щодо її правильності, такі як числова оцінка продуктивності для набору даних, а не для кожної окремої точки даних. Під час навчання НМ для задач прийняття рішень потрібно бути обережним щодо надмірного налаштування, щоб НМ вивчала минулу поведінку (або виходи) настільки точно, що вона не буде достатньо узагальненою для прогнозування майбутньої поведінки з новим набором вхідних даних.

нейронні мережі використовуються в низці дисциплін та галузей застосування²⁵⁻²⁸ У процесі прийняття рішень універсальні апроксимаційні можливості нейронних мереж (НМ) є ключовими для їхньої корисності, особливо здатність представляти нелінійне відображення між набором вхідних даних (або змінних рішення) та виходом (або рішенням). НМ часто називають «чорними скриньками», що означає, що хоча точність мережі у створенні

виходів, близьких до навчальних даних, є високою, основна інтерпретація є складною в задачах прийняття рішень. Значна частина труднощів полягає в тому, що обчислення розподілені по вузлах і, можливо, прихованих шарах, а ще більшої складності може додати надлишкові мережі. Інтерпретованість була б корисною для осіб, які приймають рішення, щоб допомогти їм зрозуміти взаємозв'язки між вхідними даними, або змінними рішення, та виходами, або рішеннями, що приймаються. Така прозорість була б також дозволяють вводити знання предметної області, такі як правила прийняття рішень, у задачу прийняття рішень як частину нейронної мережі, що, можливо, робить її ефективнішою.

Нечітка логіка як основа для інтелектуальної підтримки рішень

Хоча нейронні мережі мають здатність наближено оцінювати будь-яку неперервну функцію та навчатися, стикаючись з новими парами вхід-вихід, деякі задачі прийняття рішень мають неточні, неоднозначні або неповні вхідні дані. Нечітка логіка надає інструмент для представлення цієї невизначеності, дозволяючи вхідним даним мати значення в діапазоні значень від 0 (повністю хибно) до 1 (повністю істинно) [34]. У деяких задачах прийняття рішень корисно мати можливість представляти вхідні дані, які не є однозначно бінарними. Наприклад, температуру можна характеризувати як гарячу, холодну, теплу або прохолодну. Цей тип представлення більше відповідає людському мисленню, яке використовує відтінки значень для представлення вхідних даних та для міркування на їх основі для прийняття рішення.

Нечітка логіка надає спосіб розробки та кодування поведінки на основі правил. Оскільки знання експерта можна закодувати як набір правил, експертизу можна фіксувати та надавати особі, яка приймає рішення. Основною перевагою нечіткої логіки є те, що її можна уточнювати в міру появи нової інформації, забезпечуючи певний рівень контролю протягом усього процесу прийняття рішень. Оскільки нечітка логіка не має внутрішньої структури для вираження зв'язків між змінними в задачі прийняття рішень, нелінійні зв'язки виникають природним чином.

Нечітку логіку можна поєднувати з нейронною мережею (NN), щоб

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

інтерпретація змінних рішення була більш прозорою, вирішуючи важливу проблему при використанні NN для задач прийняття рішень. Однією з можливостей є використання параметричного представлення нечітких множин, вважаючи всі вони однакового типу. Наприклад, трикутна функція належності вимагає трьох значень для опису змінної: мінімуму, максимуму та найбільш ймовірного значення. У випадку гаусівських нечітких множин, простір представлення з двома параметрами забезпечує модальне значення та розподіл або розкид. Можна уявити зв'язок між NN та нечіткими множинами таким чином, щоб нечіткі множини описували вхідні дані рішення у формі, зрозумілій для людини. Таким чином, особа, яка приймає рішення, зможе краще сприймати відносні зв'язки між вхідними та вихідними даними, покращуючи розуміння та навчання під час прийняття рішень. Ще однією перевагою є те, що відповідну інформацію, якою володіє особа, яка приймає рішення, таку як відповідні знання предметної області, легше враховувати та використовувати.

Категорія нейронної мережі, яка є синергетичною з нечіткою логікою, називається мережею нечіткої логіки.³⁰ Обробка здійснюється логікою на рівні окремого нейрона. Агрегативні нейрони виконують І-АБО логічну агрегацію вхідних даних. Зв'язки між нейронами є важливими для навчання, оскільки вони дозволяють моделювати різні результати рішень, пов'язані з різними вхідними даними [35]. Референційні нейрони підтримують референційну або предикатну обробку, таку як *менший ніж*, *більший ніж*, *подібний*, *різний* де кожен із цих предикатів з двома аргументами виражається мовою нечіткої логіки.

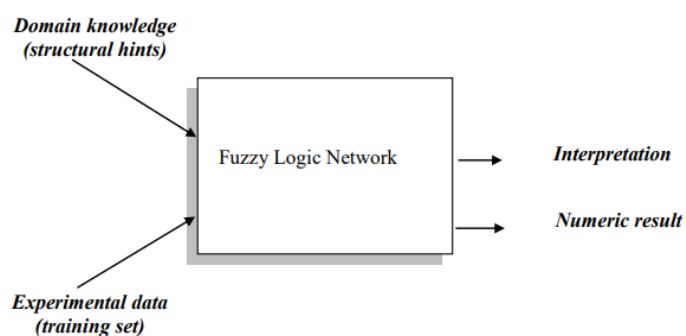


Рисунок 3.6 - Навчання та інтерпретація мереж нечіткої логіки

Мережі з нечіткою логікою допомагають усунути недоліки нейронної мережі. Оскільки архітектура прозора, знання предметної області можуть бути інтегровані в адаптивну мережу. Навчена покращити прийняття рішень, як показано на рисунку 3.6.

3.3 Застосування еволюційних алгоритмів для оптимізації рішень

Біологічні системи надихнули дослідників штучного інтелекту завдяки своїй здатності постійно вдосконалюватися, щоб адаптуватися до навколишнього середовища. Вдосконалені біологічні форми виникають та еволюціонують з попередніх поколінь, а тиск навколишнього середовища призводить до виживання найпристосованіших шляхом природного відбору. Методи штучного інтелекту, які називаються еволюційними обчисленнями, намагаються імітувати ці якості виникнення, виживання та вдосконалення, щоб адаптуватися до навколишнього середовища. Генетичні алгоритми, мабуть, є найбільш використовуваними з методів еволюційних обчислень для вирішення проблем прийняття рішень [36]. Алгоритми імітують сукупність особин, які взаємодіють та синхронізують діяльність, спілкуючись, обмінюючись локальними даними та впливаючи один на одного протягом наступних поколінь, щоб точніше відповідати навколишньому середовищу.

У більшості підходів еволюційних обчислень для задач прийняття рішень використовується стохастичний підхід, заснований на популяції. Скінчення популяція особин, представлена як елементи простору пошуку, випадковим чином ініціалізується в момент часу $t=0$. Кожна особина характеризується значенням пристосованості, пов'язаним з цільовою функцією, яка виражає вимоги середовища. Значення пристосованості відображає придатність особини до середовища [37]. Особи з вищими значеннями пристосованості мають більше шансів вижити та бути обраними батьками, які згодом формують потомство шляхом випадкової рекомбінації та мутації. Однак слабкі особини також можуть стати батьками, хоча й з меншою ймовірністю відбору. Відбір покращує загальну

якість, або пристосованість, популяції протягом наступних поколінь. Критерії зупинки можуть базуватися на кількості поколінь або статистиці, пов'язаній з функцією пристосованості (наприклад, відсутність суттєвих змін у середніх значеннях пристосованості). Ілюстрація еволюційної адаптації (або оптимізації) показана на рисунку 3.7. З часом популяція стає більш сфокусованою з точки зору пристосованості, а середня пристосованість зростає.

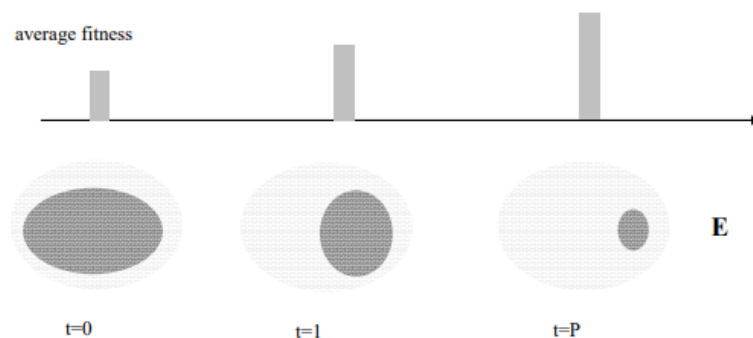


Рисунок 3.7 - Ілюстрація еволюційної оптимізації, під час якої популяція стає більш цілеспрямованою, а середня пристосованість зростає

Еволюційні обчислення пропонують привабливий метод вирішення задач прийняття рішень завдяки всебічному дослідженню простору предметної області та потенціалу для визначення глобального максимуму (за умови, що можна визначити ключові параметри алгоритму). Моделі прийняття рішень можуть отримати користь від еволюційних обчислень кількома способами:

- Зазвичай для представлення проблеми прийняття рішення доступно кілька моделей. Еволюційна оптимізація може допомогти у виборі додаткової структури для моделі прийняття рішення.

- У задачах прийняття рішень зазвичай задіяно кілька критеріїв, а еволюційні обчислення забезпечують їх одночасну оптимізацію як частину процесу розв'язання. Еволюційні обчислення пропонують низку потенційних рішень на шляху до оптимального, і ці неоптимальні рішення можуть дати розуміння проблеми, що може допомогти у прийнятті рішень.

Хоча різні методи штучного інтелекту, такі як штучні нейронні мережі,

генетичні алгоритми, міркування на основі прецедентів, методи експертних систем та представлення знань, були успішно інтегровані в інтелектуальні системи прийняття рішень, інтелектуальні агенти (ІА) мають найширше застосування для вирішення проблем прийняття рішень.³¹ Визначення інтелектуального агента різняться, хоча те, що дається Вулдріджем³² часто цитується: «Агент - це комп'ютерна система, розташована в певному середовищі та здатна до автономних дій у цьому середовищі для досягнення мети свого проектування». До цього визначення можна додати додаткові можливості, включаючи реактивність, проактивність, соціальні здібності, адаптивність, співпрацю, наполегливість та мобільність.² Реактивність та адаптивність означають, що ІА може сприймати середовище та реагувати на зміни. Проактивність означає, що ІА може ініціювати дії для досягнення мети свого проектування. Соціальні здібності та співпраця забезпечують можливості комунікації з іншими агентами, такі як переговори та співпраця. Наполегливість забезпечує здатність підтримувати стан протягом тривалих періодів часу, а мобільність дозволяє ІА подорожувати системою для отримання знань або виконання завдань.

Команди агентів можуть балансувати між пошуком цілей та реакцією на середовище, що дозволяє реалізувати складну поведінку. Багатоагентні системи (MAS), в яких кілька агентів взаємодіють один з одним, забезпечують ще більш інтелектуальну поведінку. Команди агентів можуть діяти від імені особи, яка приймає рішення, або інших агентів з різними цілями та завданнями. Успішна взаємодія базується на координації, переговорах, навчанні та довірі, особливо коли агенти не поділяють спільних переконань, цілей чи інтересів. Особи, які приймають рішення, можуть отримувати рекомендації, знаючи, що багато поглядів були збалансовані, як це передбачено системою, або навіть дозволяти системі приймати автономні рішення за певних обмежень.

Теорії агентів формують основу інтелекту [38]. Оскільки системи розподілилися по багатьох обчислювальних платформах, розподілений штучний інтелект (PAI) зосередився на багатоагентних взаємодіях у поєднанні з

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

парадигмами штучного інтелекту для розвитку високоскладного інтелекту. Сучасні дослідження безпосередньо спрямовані на інтеграцію людини в системи, щоб агенти та люди могли взаємодіяти та навчатися один в одного. Основні поняття агентної технології описуються так:

Комунікація: ІА повинні спілкуватися, щоб передати свої наміри. Хоча комунікація є невід'ємною частиною взаємодії, вона не обов'язково має бути прямою. Непряма комунікація може відбуватися шляхом спостереження за діями. У MAS комунікація може бути реалізована за допомогою передачі повідомлень або спільних змінних.

Координація: Координація – це засіб організації агентів, їхніх ресурсів та завдань з метою вирішення конфліктів та покращення їхньої продуктивності.

Угоди про командну роботу: Для взаємодії будь-якого ІА з іншим ІА та діяння як команди необхідні комунікація та координація. Такі домовленості про командну роботу, як співпраця та колаборація, є підпоняттями, що описують тип угоди між командою ІА. Людиноцентричні агенти: новіші дослідження розглядали взаємодію між ІА та людиною, завдяки чому особа, яка приймає рішення, стає частиною команди.

Навчання: Навчання як атрибут робить MAS більш схожою на людину та збільшує можливості для успішної взаємодії між людьми та агентами. Гібридні стратегії, що поєднують традиційні методи ШІ з моделями міркування з когнітивної науки, такими як модель Саймона,³запропонувати нові способи покращення підтримки прийняття рішень.

Однією з фреймворків, яка привернула увагу для розробки складних реактивних систем, таких як багатоагентна командна робота або командна робота людина-агент, є фреймворк «Переконання-Бажання-Намір» (BDI) [39]. Розуміння агентом зовнішнього світу – це його переконання; цілі, яких він повинен досягти, – це його бажання; а напрямки дій, яких агент зобов'язаний дотримуватися для задоволення своїх бажань, – це його наміри. Включення зовнішнього світогляду може забезпечити усвідомлення ситуації або контекст для прийняття рішень. Контекст, або «те, що обмежує щось, не втручаючись у це

явно», вважається важливим для складних застосувань прийняття рішень. Хоча абстрактно його важко кількісно оцінити, контекст можна охарактеризувати як відносний до фокусу уваги, що розвивається разом із фокусом уваги та дуже залежить від предметної області. Нові рамки для командної роботи в MAS та командах людина-агент підтверджують поняття контексту в проблеми з рішеннями.

Для полегшення взаємодії між людьми та машинами для спільного прийняття рішень можна створювати «віртуальних людей», які не лише виглядають як люди, але й демонструють автономію, взаємодію та персоніфікацію. Ці віртуальні люди втілюють такі характеристики, як емоції, і можуть виражати їх через реакції, такі як міміка, що збільшує потенціал для сприйняття людиною машинного мислення.³³ Такі потужні концепції спонукають нас прийняти людські обмеження та розширити наші можливості прийняття рішень у проблемах, які можуть виграти від впровадження інструментів штучного інтелекту.

Майбутнє відкриває дослідницькі можливості та виклики у застосуванні інструментів штучного інтелекту для підтримки прийняття рішень, зокрема в інтерфейсах між людьми та машинами. Можливості для покращення процесу прийняття рішень є значними, особливо у складних проблемах, де середовище перевищує наші можливості для розуміння та розвитку взаємозв'язків між змінними. Завдання полягає в розробці інтелектуальних систем підтримки прийняття рішень, які є економічно ефективними, забезпечують відчутні переваги та дають результати, прийнятні для людей [40]. Активно досліджуються адаптивні системи, які персоналізуються для різних користувачів та сприймають наміри користувача в діях чи мові.

Однією з найбільших проблем у застосуванні інтелектуальних систем підтримки рішень для вирішення реальних проблем є довіра до автономних систем. Майбутні дослідження повинні будуть відповісти на такі питання, як: Які рішення ми готові дозволити комп'ютерним системам приймати автономно? Які докази точності нам потрібні, щоб призначити рішення автономній системі? Чи

дозволимо ми автономним системам приймати рішення та діяти відповідно до цих рішень, і за яких умов? Який рівень безпеки потрібен, щоб комп'ютерні системи не перевищували наш рівень комфорту зі своїми рішеннями? Чи справді ми довіряємо автономним системам діяти в наших інтересах? Досягнення в інструментах штучного інтелекту, що застосовуються для підтримки рішень, пропонують захопливі можливості для покращення процесу прийняття рішень та вирішення цих питань.

3.4 Висновки по розділу

У третьому розділі було розглянуто практичну реалізацію та оцінку ефективності інтелектуальних систем підтримки прийняття рішень. Проведено моделювання процесів прийняття рішень у програмних системах, що дозволило формалізувати та оптимізувати логіку вибору рішень. Досліджено використання нейронних мереж для підвищення точності прогнозування та аналізу даних у СППР. Також розглянуто застосування еволюційних алгоритмів для оптимізації рішень у складних багатокритеріальних задачах. Проведений аналіз результатів підтвердив ефективність використання інтелектуальних методів для підвищення якості та обґрунтованості прийняття рішень.

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

ВИСНОВКИ

У ході виконання бакалаврської роботи було досліджено та систематизовано підходи до створення програмної підтримки інтелектуальних систем прийняття рішень, що є складним багаторівневим процесом, спрямованим на підвищення ефективності аналізу даних і якості управлінських рішень. У роботі проаналізовано сучасні підходи до побудови систем підтримки прийняття рішень, розглянуто роль штучного інтелекту, а також досліджено методи інтеграції інтелектуальних компонентів у програмні системи. Отримані результати дозволили сформулювати цілісне бачення архітектури та функціонування інтелектуальних СППР.

На початковому етапі було проведено аналіз предметної області, що включав дослідження теоретичних основ систем підтримки прийняття рішень, їх еволюції та взаємозв'язку зі штучним інтелектом. Особливу увагу приділено аналізу індивідуальних та групових СППР, а також їх здатності адаптуватися до змін середовища. Це дозволило визначити ключові вимоги до сучасних інтелектуальних систем, серед яких гнучкість, масштабованість, здатність до навчання та обробки великих обсягів даних.

У другому розділі було досліджено методи та інструменти розробки інтелектуальних систем прийняття рішень. Розглянуто процеси проєктування таких систем, інтеграцію підходів системної інженерії та прийняття рішень, а також сучасні моделі розробки програмного забезпечення. Значну увагу приділено аналізу архітектурних рішень, що забезпечують ефективну взаємодію між компонентами системи, а також оптимізацію продуктивності та надійності.

У третьому розділі було розглянуто практичні аспекти реалізації інтелектуальних СППР. Зокрема, досліджено застосування нейронних мереж як інструменту аналізу та прогнозування, а також використання еволюційних алгоритмів для оптимізації процесу прийняття рішень. Проведено аналіз ефективності запропонованих підходів, що дозволило оцінити їхню доцільність у різних прикладних сценаріях. Отримані результати свідчать про високу ефективність використання інтелектуальних методів у складних задачах

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

прийняття рішень.

У процесі роботи було визначено ключові переваги інтелектуальних систем прийняття рішень, серед яких автоматизація аналізу, підвищення точності прогнозування, зменшення впливу людського фактору та можливість роботи з великими обсягами даних. Водночас виявлено і певні обмеження, такі як складність реалізації, необхідність значних обчислювальних ресурсів та потреба у якісних навчальних даних.

Загалом, результати роботи підтверджують, що програмна підтримка інтелектуальних систем прийняття рішень є перспективним напрямом розвитку програмної інженерії. Використання сучасних методів штучного інтелекту дозволяє значно підвищити ефективність прийняття рішень у різних галузях. У подальшому можливе розширення функціональності таких систем за рахунок інтеграції з великими даними (Big Data), хмарними технологіями та системами реального часу, що відкриває нові можливості для їх практичного застосування.

					БР.ІІІ – 05.00.00.000 ПЗ	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА

1. Turban, E., Sharda, R., & Delen, D. (2021). Decision Support and Business Intelligence Systems. pearson.com. <https://www.pearson.com>
2. Power, D. J. (2020). Decision Support Systems: Concepts and Resources. dssresources.com. <https://dssresources.com>
3. Marakas, G. M. (2003). Decision Support Systems in the 21st Century. prenticehall.com.
4. Keen, P., & Scott Morton, M. (1978). Decision Support Systems: An Organizational Perspective. addison-wesley.com.
5. Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach. pearson.com. <https://www.pearson.com>
6. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. mitpress.mit.edu. <https://www.deeplearningbook.org>
7. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. springer.com.
8. Mitchell, T. (1997). Machine Learning. mcgraw-hill.com.
9. Haykin, S. (2009). Neural Networks and Learning Machines. pearson.com.
10. Holland, J. H. (1992). Adaptation in Natural and Artificial Systems. mitpress.mit.edu.
11. Sommerville, I. (2020). Software Engineering. pearson.com.
12. Pressman, R. S. (2019). Software Engineering: A Practitioner's Approach. mcgraw-hill.com.
13. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design Patterns. pearson.com.
14. Fowler, M. (2002). Patterns of Enterprise Application Architecture. martinowler.com. <https://martinfowler.com>
15. Martin, R. C. (2017). Clean Architecture. pearson.com.

16. Microsoft Azure AI Documentation. (2025). learn.microsoft.com.
<https://learn.microsoft.com/en-us/azure/ai-services/>
17. Azure Machine Learning Documentation. (2025). learn.microsoft.com.
<https://learn.microsoft.com/en-us/azure/machine-learning/>
18. Google AI Platform Documentation. (2025). cloud.google.com.
<https://cloud.google.com/ai>
19. IBM Watson AI Documentation. (2025). ibm.com.
<https://www.ibm.com/watson>
20. TensorFlow Documentation. (2025). tensorflow.org.
<https://www.tensorflow.org>
21. PyTorch Documentation. (2025). pytorch.org. <https://pytorch.org>
22. Scikit-learn Documentation. (2025). scikit-learn.org. <https://scikit-learn.org>
23. Keras Documentation. (2025). keras.io. <https://keras.io>
24. RapidMiner Platform Guide. (2024). rapidminer.com.
<https://rapidminer.com>
25. KNIME Analytics Platform. (2024). knime.com. <https://www.knime.com>
26. OWASP Top Ten Security Risks. (2024). owasp.org.
<https://owasp.org/www-project-top-ten/>
27. ISO/IEC 25010:2011 Systems and Software Quality Requirements.
iso.org. <https://www.iso.org>
28. IEEE Software Engineering Standards. (2023). ieee.org.
<https://www.ieee.org>
29. Gartner. (2024). Trends in AI-driven decision support systems.
gartner.com. <https://www.gartner.com>
30. Deloitte. (2025). AI-powered decision-making systems. deloitte.com.
<https://www2.deloitte.com>
31. Chen, H., Chiang, R., & Storey, V. (2012). Business Intelligence and Analytics. MIS Quarterly, 36(4), 1165–1188. <https://doi.org/10.2307/41703503>

32. Shim, J. P., Warkentin, M., Courtney, J. F., et al. (2002). Past, present, and future of DSS. Decision Support Systems, 33(2), 111–126. [https://doi.org/10.1016/S0167-9236\(01\)00139-7](https://doi.org/10.1016/S0167-9236(01)00139-7)
33. Power, D. J., & Sharda, R. (2007). Model-driven DSS. Decision Support Systems, 43(3), 1044–1061. <https://doi.org/10.1016/j.dss.2005.05.030>
34. Wang, L., & Zhang, Y. (2025). Intelligent decision support with AI. ACM Transactions on Information Systems, 41(2), 1–20. <https://doi.org/10.1145/3651234>
35. Li, X., & Chen, J. (2024). Neural networks for decision support. Expert Systems with Applications, 235, 120123. <https://doi.org/10.1016/j.eswa.2024.120123>
36. Kumar, S., & Sharma, R. (2023). AI-based DSS architectures. Journal of Software Engineering, 11(2), 67–82. <https://doi.org/10.1186/s40411-023-00189-9>
37. Taylor, M., & Brown, A. (2025). Security in intelligent DSS. IEEE Security & Privacy, 23(1), 45–53. <https://doi.org/10.1109/MSEC.2024.347890>
38. Oracle AI and Analytics Documentation. (2025). oracle.com. <https://www.oracle.com/artificial-intelligence/>
39. SAP Analytics Cloud Documentation. (2025). sap.com. <https://www.sap.com/products/technology-platform/cloud-analytics.html>
40. □ Dataversity. (2024). Data-driven decision support systems. dataversity.net. <https://www.dataversity.net>

БІБЛІОГРАФІЧНА ДОВІДКА

Тема бакалаврської роботи: " Програмна підтримка інтелектуальних систем прийняття рішень "

Обсяг пояснювальної записки: 63 аркушів

Дата закінчення дипломної роботи 10 червня 2026р.

Підпис студента _____