**Міністерство освіти і науки України**
**Національний аерокосмічний університет**
**ім. М. Є. Жуковського «ХАІ»**

# МЕТОДИ ТА ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ТА БЕЗПЕКИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

**Монографія**

**За редакцією В. С. Харченка, О. І. Морозова**

# METHODS AND TECHNOLOGIES OF ENSURING QUALITY AND SAFETY OF INTELLIGENT SYSTEMS

**Monograph**

**Edited by V. S. Kharchenko, O. I. Morozova**

**Проєкти**
***Методологія та інформаційні технології оцінювання та забезпечення безпеки цифрової інфраструктури малих модульних реакторів (Д 503-4/2022-Ф, № Д/Р 0122U000977)***

***Методи, програмно-апаратні засоби та технології забезпечення гарантоздатності інтелектуальних систем індустріального інтернету речей (Д 503-10/2022-П, № Д/Р 0122U001065)***

**Харків – 2023**

UDC 004.8/.9.05(02)
M54

The monograph is based on the research results in area of methods and techniques for assessing and providing safety and security intelligent mobile systems (IMS) and systems of industrial Internet of Things (IIoT) that were obtained by author's team of the Computer Systems, Networks and Cyber Security Department, National Aerospace University «Kharkiv Aviation Institute», researchers of other universities and industrial enterprices. It is devoted to the analyzing and developing principles, models, methods, and technologies of designing safe and secure IMSs and IIoT. The book is published with support of the projects № D/R 0122U000977, № D/R 0122U001065 funded by Ministry of Education and Science of Ukraine.

This book is intended for MSc- and PhD-students, university lecturers, engineers, and researchers in the area of intelligent systems and technologies, safety and security of critical infrastructures.

Ref. – 516 items, figures –112, tables – 56.

**Рецензенти**: доктор технічних наук, професор **Опанасенко Володимир Миколайович**, Інститут кібернетики ім. В. М. Глушкова НАН України;
доктор технічних наук, професор **Заславський Володимир Анатолійович**, Київський національний університет імені Тараса Шевченка.

Методи та технології забезпечення якості та безпеки інтелектуальних систем : кол. моногр. / за ред. В. С. Харченка, О. І. Морозової. – Міністерство освіти і науки України, Національний аерокосмічний університет ім. М. Є. Жуковського «ХАІ». – К. "Видавництво "Юстон", 2023. – 352 с.

ISBN 978-617-8335-01-4

Монографія базується на результатах досліджень методів і засобів оцінювання та забезпечення безпеки інтелектуальних мобільних систем (ІМС) і систем індустріального інтернету речей (ІІР), які виконано колективом авторів кафедри комп'ютерних систем, мереж і кібербезпеки Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут» та інших університетів. Присвячена аналізу та розвитку принципів, моделей, методів та технологій побудови безпечних ІМС та ІІР. Монографія видана за підтримки проєктів № Д/Р 0122U000977, № Д/Р 0122U001065, які фінансуються Міністерством освіти і науки України.

Для студентів, аспірантів і викладачів університетів, інженерів та дослідників у сфері безпечних інтелектуальних систем і технологій, а також безпеки критичних інфраструктур.

Бібл. –516 найменувань, рисунків – 112, таблиць – 56.

Монографія рекомендована до видання Вченою радою Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут» (протокол № 4 від 25 листопада 2022 року).

УДК 004.8/.9.05(02)



ISBN 978-617-8335-01-4

ЗМІСТ

# АНОТАЦІЯ

Монографія базується на результатах досліджень у галузі безпеки інтелектуальних мобільних систем (ІМС) та систем індустріального інтернету речей (ІІР), які виконано колективом авторів кафедри комп'ютерних систем, мереж і кібербезпеки Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», а також інших університетів та індустріальних підприємств України, Греції, Ізраїля. Присвячена аналізу та розвитку моделей, методів і технологій побудови та та забезпечення безпеки та гарантозхдатності ІМС та ІІР. Складається з 18 розділів, які об'єднано в три частини:

- моделі та методи оцінювання якості інтелектуальних систем;
- методи і технології забезпечення безпеки інтелектуальних систем для мобільних та індустріальних комплексів;
- методи і технології побудови безпечних інтелектуальних систем.

Монографія видана за підтримки проєктів, які фінансуються Міністерством освіти і науки України:

- Методологія та інформаційні технології оцінювання та забезпечення безпеки цифрової інфраструктури малих модульних реакторів (Д 503-4/2022-Ф, № Д/Р 0122U000977);

- Методи, програмно-апаратні засоби та технології забезпечення гарантоздатності інтелектуальних систем індустріального інтернету речей (Д 503-10/2022-П, № Д/Р 0122U001065).

Для студентів, аспірантів і викладачів університетів, інженерів та дослідників у сфері інтелектуальних систем і технологій, а також безпеки мобільних систем, індустріального інтернету речей та критичних інфраструктур.

***60-річчю кафедри комп'ютерних систем, мереж і кібербезпеки Національного аерокосмічного університету «ХАІ» присвячується***

## ВСТУП

### 0.1. Передумова і задачі

Безпекові засоби, штучний інтелект, індустріальний інтернет речей та мобільні системи є найбільш важливими, критичними та динамічними напрямами розвитку сучасних технологій, які впливають на економіку, оборону, повсякденне життя. Це обумовлює актуальність досліджень, результати яких представлено в монографії, оскільки вони одержані на перетині згаданих напрямів.

Дослідження виконувалися науковцями, викладачами, аспірантами, магістрантами кафедри компютерних систем, мереж і кібербезпеки, Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут», а також колегами з інших університетів та індустріальних підприємств впродовж останніх п'яти років в рамках низки міжнародних і національних проектів, зокрема, проєктів № Д/Р 0122U000977, № Д/Р 0122U001065.

Метою даного видання є представлення наукових результатів за напрямами:

- розроблення та дослідження моделей та методів оцінювання якості інтелектуальних систем;
- розроблення методів і технологій забезпечення безпеки інтелектуальних систем для мобільних та індустріальних комплексів;
- розроблення методів і засобів побудови безпечних інтелектуальних систем.

### 0.2. Структура

Монографія складається з трьох частин, які об'єднують 18 розділів з типовою структурою і окремим списком літератури.

Перша частина присвячена розробленню та дослідженню моделей та методів оцінювання якості інтелектуальних систем і об'єднує п'ять розділів:

1. Принципи та моделі якості та профілювання вимог до інтелектуальних систем.
2. Метод і моделі метричного оцінювання якості інтелектуальних систем.
3. Методи оцінювання пояснюваного штучного інтелекту як сервісу.
4. Методи оцінювання якості систем доповненої реальності.
5. Методи оцінювання та підвищення якості інформаційних систем на підставі аналізу даних.

Друга частина об'єднує матеріали розділів 6-9, які описують методи і технології оцінювання та забезпечення безпеки інтелектуальних систем:

6. Аналіз методів і засобів оцінювання і забезпечення кібербезпеки систем і сервісів штучного інтелекту.

7. Аналітичні та експериментальні методи оцінювання функційної та кібербезпеки робототехнічних систем.

8. Методи забезпечення кіберббезпеки веб-систем з використанням засобів штучного інтелекту.

9. Розробка моделі загроз для безпілотних літальних апаратів.

Третя частина описує результати досліджень методів і технологій побудови безпечних інтелектуальних систем, які описано в розділах 10-18:

10. Інтелектуальні системи моніторингу потенційно небезпечних обєктів з використанням технологій літаючих крайових обчислень.

11. Інтелектуальні системи інтернету речей з використанням технологій цифрових двійників.

12. Технології інтернета речей для побудови безпечних веб-орієнтованих систем.

13. Методи і технології побудови та дистанційної реконфігурації вузлів вбудованої системи.

14. Знання-орієнтовані методи та засоби автоматизації синтезу тестів.

15. Методи семантичної кластеризації даних для застосування технологій доповненої реальності в інтерактивному мистецтві.

16. Методи пошуку та ідентифікації вибухо-небезречних предметів з використанням багатоцільових інтелектуальних безпілотних систем.

17. Методи і засоби штучного інтелекту для надійного керування мобільними системами освітлення.

18. Ефективність використання явних та неявних користувацьких фідбеків для гібридних рекомендаційних систем.

Автори висловлюють щиру подяку колективу кафедри комп'ютерних систем, мереж і кібербезпеки Національного аерокосмічного університету ім. М. Є. Жуковського «Харківський авіаційний інститут», а також всім колегам, які долучилися до обговорення результаів досліджень. Ця книга присвячується 60-річчю кафедри, яка має багаторічну історію і була однією з перших кафедр обчислювальної техніки і програмування в Україні.

Окрема подяка рецензентам – д.т.н., професору Опанасенку Володимиру Миколайовичу, Інститут кібернетики ім. В. М. Глушкова НАН України, і д.т.н., професору Заславському Володимиру Анатолійовичу, Київський національний університет імені Тараса Шевченка.

Редактори: В. С. Харченко, д.т.н., професор, лауреат Державної премії України в галузі науки і техніки, заслужений винахідник України, завідувач кафедри компютерних систем, мереж і кібербезпеки; О. І. Морозова, д.т.н., професор, професор кафедри комп'ютерних систем, мереж і кібербезпеки.

# ЧАСТИНА І. МОДЕЛІ ТА МЕТОДИ ОЦІНЮВАННЯ ЯКОСТІ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

## 1. ПРИНЦИПИ ТА МОДЕЛІ ЯКОСТІ ТА ПРОФІЛЮВАННЯ ВИМОГ ДО ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

**О. О. Ілляшенко, О. І. Морозова, Г. В. Фесенко, В. С. Харченко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

### 1.1. Вступ

Мотивація. Якість життя, безпека окремих людей і, навіть, країн залежать від інформаційних технологій, серед яких найбільш складними і дещо суперечливими є технології штучного інтелекту (ШІ). Динаміка впровадження систем ШІ (СШІ) в різних сферах, інтенсивні розробки і дослідження супроводжуються стрімким збільшенням кількості публікацій за останні три роки [1], численних технічних звітів і стандартів європейських інституцій [2, 3], ISO/IEC [4-6], IEEE [7], NIST [8-10], OECD [11], UNESCO [12].

В індустріальних системах, медицині, транспорті, системах озброєння, юриспруденції тощо вплив ШІ стає, з одного боку, все більш відчутним і сталим, а з іншого, - вельми суперечливим, що обумовлено кількома чинниками:

- складністю рішень, які приймаються при розробленні та застосуванні систем, в які вбудовано засоби ШІ;
- змінним та не завжди визначеним фізичним та інформаційним середовищем, в якому вони функціонують СШІ. Зростає інтенсивність і розширюється номенклатура зовнішніх впливів, кібератак, які, з одного боку, спрямовані на штучний інтелект, з іншого, – базуються на методах ШІ;
- накопиченням експертної інформації та розширенням баз знань, які можуть бути використані для підвищення ефективності СШІ. Принцип людиноцентричності при їх створенні та застосуванні має бути збалансованим задля мінімізації ризиків прийняття помилкових рішень внаслідок суб'єктивних причин;
- зростанням ваги етичних і безпекових аспектів впродовж використання. Цей чинник є особливо важливим і специфічним для СШІ. Відповідно до [12] та інших документів, які зосереджуються на гуманітарних аспектах, людська гідність, персональна і колективна безпека та благополуччя є ціннісними орієнтирами при розробленні та впровадженні систем ШІ.

Означені обставини ускладнюють, на відміну від «традиційних» систем, формулювання специфікацій та перевірку виконання вимог при створенні та модернізації СШІ. Крім того, зростає кількість і різноманітність характеристик ШІ та СШІ, які мають бути враховано, зокрема, таких як етичність,

пояснюваність, довірчоздатність тощо [13-15]. В свою чергу, урізноманітнюються методи оцінювання, що мають базуватися на чіткому уявленні про сутність і взаємозалежність характеристик ШІ.

Слід підкреслити, що зростання кількості публікацій і стандартів супроводжується суттєвою невпорядкованістю характеристик ШІ, яка, з одного богу, обумовлює, з іншого, - обумовлюється неузгодженістю їх визначень. Отже, вкрай важливими є дослідження задля гармонізації та ієрархізації характеристик, що надасть об'єктивності і спростить розроблення інструментарію нормування, оцінювання та забезпечення вимог при створенні та впровадженні СШІ.

Мета і структура. Метою дослідження є розроблення моделі якості штучного інтелекту на основі визначення та упорядкування характеристик. Ці характеристики називаємо нефункційними за аналогією з характеристиками програмного забезпечення та ІТ-систем, оскільки вони є загальними для різних застосунків. Задачі дослідження полягають у наступному:

- сформулювати принципи та обґрунтувати послідовність аналізу і розроблення моделей якості ШІ як впорядкованих множин характеристик;
- запропонувати моделі якості ШІ задля подальшого використання, перш, за все, оцінювання окремих характеристик і якості в цілому;
- продемонструвати варіанти профілювання моделей якості ШІ для систем моніторингу інженерних комунікацій і розпізнавання дорожніх знаків.

Стаття структурується у такий спосіб. В наступному розділі обґрунтовуються принципи та послідовність розроблення моделей якості. В третьому підрозділі пропонуються підходи щодо формулювання визначень характеристик ШІ на підставі аналізу існуючих та їх гармонізації з урахуванням різних груп джерел, а також надається таблиця з визначеннями і класифікацією характеристик якості ШІ. Наступний підрозділ описує загальну модель ШІ з наданням опису покрокової процедури реалізації її ієрархічної побудови; представлено так звану базову модель зі скороченими множинами характеристик з огляду на їх важливість. У наступному підрозділі описуються приклади моделей якості для двох систем штучного інтелекту. Останній підрозділ надає висновки і описує напрямки подальших досліджень.

## 1.2. Принципи та послідовність досліджень

Сукупність характеристик СШІ, які аналізуються в статті, об'єднуються поняттям «якість» за аналогією з тим, як це зазвичай робиться для програмного забезпечення, де існують сталі моделі якості, що розвивалися і удосконалювалися впродовж майже 55 років еволюції [16-18]. Поняття «якість» ШІ, на наш погляд, є прийнятною узагальнюючою характеристикою, не зважаючи на те, що деколи її використовують як часткову характеристику СШІ або розглядають якість штучного інтелекту суто в контексті якості програмного забезпечення [19].

Контекст якості програмного забезпечення є дійсно дуже важливим, але він має використовуватися як підхід для формування більш загальної моделі якості ШІ. В [20] формується думка, аналогічна позиції авторів даного дослідження щодо важливості саме якості АІ. Однак робота [20] дещо звужує зміст якості, оскільки набір характеристик ШІ, які аналізуються, є обмеженим. Отже надалі ми використовуємо поняття якості ШІ як системоутворюючої, верхньорівневої сутності в ієрархії всіх характеристик згідно із загальним тлумаченням якості за стандартом ISO 9001:2015, тобто ступеня, в який набір властивих об'єкту (в даному випадку ШІ) характеристик відповідає вимогам.

Якість СШІ складається з якості власне штучного інтелекту як узагальненого об'єкту і якості програмно-апаратної платформи, за допомогою якої ШІ реалізується. Це дослідження розглядає складові якості (характеристики) тільки ШІ.

Ключовим поняттям, яке використовується в дослідженні, є «характеристика» - складова якості, що описує різні властивості ШІ. Характеристика є базою для формулювання вимог до системи штучного інтелекту та її компонентів шляхом:

- урахування відповідної характеристики при розробленні специфікації – переліку вимог до СШІ;
- визначення метрик, за допомогою яких оцінюється значення характеристики, а саме, шкали і методики вимірювань (оцінювання);
- обґрунтування необхідних «меж» цієї характеристики, тобто вимог до її якісного або кількісного рівня, що визначаються відповідними метриками.

Якщо характеристика штучного інтелекту ChAI-1 є залежною від характеристики ChAI-2, то ChAI-2 називатимемо підхарактеристикою характеристики ChAI-1. Відповідно ChAI-1 і ChAI-2 мають розташовуватися на верхньому та наступному нижньому рівнях ієрархії моделі якості. Для кожної з характеристик (підхарактеристик) мають бути визначені метрики для їх оцінювання, а також сформульовано вимоги до значень характеристик, розроблено профіль вимог та оцінено його якість [22].

Послідовність побудови моделей якості ШІ є такою: на підставі аналізу посилань формується список характеристик ШІ і здійснюється гармонізація їх визначень. Результатом є таблиця 1.1 (підрозділ 1.3); далі пропонуються моделі якості ШІ у графовій формі і надаються приклади профілювання моделей якості для двох систем – моніторингу інженерних комунікацій і розпізнавання дорожніх знаків (підрозділи 1.5.1 і 1.5.2).

### 1.3. Відбір і гармонізація визначень характеристик

Відбір і гармонізацію визначень характеристик ШІ було виконано наступним чином.

1. При аналізі визначень враховувалося, що окремі характеристики можуть бути тотожними, тобто такими, що мають різну назву, але однакову сутність. З підмножин таких характеристик залишалася одна для подальшого використання. Наприклад, з поміж характеристик «explicability» та «explainability», які означають «пояснюваність» (ability to be explained), для подальшого розгляду обрана характеристика «explainability».

2. Деякі характеристики з несуттєвою відмінністю були об'єднані, а у відповідних визначеннях ці відмінності враховувалися. Наприклад, характеристика «людський нагляд і рішучість» («human oversight and determination») була поглинута характеристикою «людський нагляд» («human oversight») з урахуванням особливостей проведення нагляду за ШІ, запропонованих в поглинутій характеристиці, у кінцевому визначенні.

3. Кілька характеристик були виключені, оскільки вони не мали специфічних ознак для ШІ, а є загальними для технічних систем або їх програмно-апаратного забезпечення. До таких характеристик, зокрема, належать «впевненість» («confidence») та «відповідність» («compliance»).

4. Для характеристик, які є суттєвими для ШІ, визначення надано шляхом:

- повторення (цитування) або несуттєвого коригування визначення з одного з документів, яке є найбільш адекватним і точним, на думку авторів (позначено літерою R – referred). Наприклад, визначення характеристики «цілісність» («integrity») була подано у відповідності з [21];
- гармонізації визначення на підставі визначень, які надаються в різних публікаціях (позначено літерою H – harmonized). Сутність гармонізації полягала у виявленні ключових термінів і поєднанні суттєвих складових різних визначень характеристики, що аналізувалася. Наприклад, визначення характеристики «цілісність» («integrity») було отримано шляхом поєднання суттєвих складових визначень цієї характеристики, запропонованих у [5, 21, 23];
- визначення, яке надано авторами у разі відсутності або незадовільного на їх думку формулювання для характеристики в доступних джерелах (позначено літерою A – authored). Наприклад, у такий спосіб було отримано визначення характеристики «resiliency» («резильєнтність»).

Результати аналізу джерел [1-3, 5, 6, 8-15, 20, 21, 23-36] і гармонізації визначень характеристик штучного інтелекту надано в таблиці 1. Таким чином, було відібрано 32 характеристики. Визначення чотирьох характеристик вибрано з відповідних джерел без змін; визначення 25 характеристик було гармонізовано, визначення трьох характеристик є авторським.

Таблиця 1.1 – Результати аналізу і гармонізації визначень характеристик штучного інтелекту

| № з/п | Назва характеристики | Визначення | отриман | Джерело |
|---|---|---|---|---|
| 1 | Верифікованість (verifiability, VFB) | здатність ШІ, яка характеризується ступенем пристосованості до проведення верифікації різними методами. | H | [36] |
| 2 | Відповідальність (responsibility, RSP) | здатність ШІ функціонувати з урахуванням очікувань замовника (користувача) у відповідності до етичних норм, законодавчих нормативно-правових актів, а також інформувати його у разі можливого їх порушення. | H | [12, 26, 31] |
| 3 | Відстежуваність (accountability, ACN) | здатність ШІ надавати звіти за визначеною формою про результати функціонування у прозорий спосіб. | R | [21, 27] |
| 4 | Відшкодовуваність (redress, RDR) | здатність ШІ надавати доступні механізми забезпечення адекватного відшкодування наслідків негативного впливу на людей. | H | [2, 3] |
| 5 | Диверсність (diversity, DVS) | здатність ШІ мінімізувати ризик невиконання специфікованих (визначених за необхідністю) функцій або завдань внаслідок відмов, обумовлених фізичними та інформаційними чинниками, з використанням різних моделей, алгоритмів та інших засобів. | A | [6] |
| 6 | Довірчоздатність (trustworthiness, TST) | здатність ШІ, яка характеризується ступенем впевненості користувача або іншої зацікавленої особи (розробника, аудитора тощо) в тому, що ШІ відповідає вимогам і виконує функції у передбачуваний спосіб. | H | [5, 10, 15 23] |
| 7 | Етичність (ethics, ETH) | здатність ШІ відповідати діючим нормам моралі за результатами функціонування. | H | [2, 3] |
| 8 | Завершеність (completeness, CMT) | здатність ШІ бути цілісним з точки зору ступеня відповідності всім вимогам замовника. | H | [21] |
| 9 | Законність (lawfulness, LFL) | здатність ШІ відповідати законодавчим і нормативно-правовим актам. | R | [2, 3] |

| | | | | |
|---|---|---|---|---|
| 10 | Захищеність (security, SCR) | здатність ШІ захищати інформаційні та фізичні активи таким чином, щоб інші невизначені (неавторизовані) особи чи системи, включаючи ШІ, не мали б доступу до них або мали б такий доступ відповідно до визначеного типу і рівня авторизації. | H | [12, 21, 27] |
| 11 | Зміщеність (bias, BIS) | характеристика ШІ, яка визначає ризики появи результатів, які упереджені через хибні припущення та помилки в процесі налаштування моделей (наприклад, машинного навчання). | H | [2, 3, 6, 9] |
| 12 | Зрозумілість (comprehensibility, CMH) | здатність ШІ забезпечувати для користувача (або полегшувати користувачеві) розуміння пояснень, достатніх для того, щоб надати змогу застосувати ШІ або інформацію, отриману за його допомогою, для виконання інших завдань. | A | [29] |
| 13 | Інтерактивність (interactivity, INR) | здатність ШІ забезпечувати ефективну і проактивну взаємодію з користувачем. | H | [25] |
| 14 | Інтерпретабельність (interpretability, INP) | здатність ШІ надавати та інтерпретувати інформацію у зрозумілий для користувача спосіб. | H | [32] |
| 15 | Людська автономність (human agency, HMA) | здатність ШІ надавати користувачу можливість приймати автономні обґрунтовані рішення щодо застосування ШІ. | H | [2, 3] |
| 16 | Людський нагляд (human oversight, HMO) | здатність ШІ надавати можливості користувачу контролювати і при необхідності втручатися визначеним чином в функціонування ШІ. | H | [2, 3] |
| 17 | Недискримінативність (non-discrimination, NDS) | здатність ШІ забезпечувати виконання етичних норм щодо відсутності дискримінації за будь-якими ознаками. | H | [2, 3, 12] |
| 18 | Об'єктивність (objectivity, OBC) | здатність ШІ запобігати використанню скомпроментованих або сфальсифікованих даних. | R | [33] |

Продовження табл. 1.1

| | | | | |
|---|---|---|---|---|
| 19 | Пояснюваність (explainability, EXP) | здатність ШІ бути зрозумілим і передбачуваним з точки зору призначення та поведінки. | Н | [1-3, 8, 31] |
| 20 | Приватність (privacy, PRV) | здатність ШІ забезпечувати право розпоряджатися особистою інформацією у відповідності до вимог користувача. | Н | [2, 3, 20, 33] |
| 21 | Прийнятність (acceptability, ASP) | здатність ШІ забезпечувати хоча б часткову його відповідність вимогам замовника або очікуванням споживача | Н | [24] |
| 22 | Причинність (causability, CSL) | здатність ШІ визначати причино-наслідкові зв'язки між подіями, що виникають під час його застосування. | Н | [28] |
| 23 | Простежуваність (traceability, TRC) | здатність ШІ простежувати виконання вимог у зручний для користувача спосіб, здійснювати пошук та документування помилок, невідповідностей на кожному етапі життєвого циклу. | Н | [2, 3, 34] |
| 24 | Резильєнтність (resiliency, RSL) | здатність ШІ продовжувати функціонування в умовах зміни вимог, параметрів фізичного та інформаційного середовища, а також виникнення неспецифікованих порушень і відмов. | А | [27] |
| 25 | Робастність (robustness, RBS) | здатність ШІ коректно працювати в широкому діапазоні вхідних даних та умов експлуатації і переходити у стан призупинення системи у разі виходу цих даних і умов за специфіковані межі. | Н | [2, 3, 11, 20] |
| 26 | Соціальне благополуччя (societal well-being, SWB) | здатність ШІ враховувати соціальні процеси і не шкодити фізичному та психічному почуттю людей та благополуччю суспільства в цілому. | Н | [2, 3] |
| 27 | Справедливість (fairness, FRN) | здатність ШІ мінімізувати ризики аномалій, обумовлених упередженістю при прийнятті рішень, які пов'язані з виконанням етичних норм (включаючи відсутність фаворитизму, дискримінацію за релігійними, расовими та іншими ознаками, тощо), а також хибних припущень і помилок в процесі налаштування моделей. | Н | [2, 3, 11] |

| | | | | |
|---|---|---|---|---|
| 28 | Сприйнятливість (graspability, GRS) | здатність ШІ забезпечувати можливості користувачу критичного сприйняття ШІ в рамках відкритого і демократичного середовища. | H | [30] |
| 29 | Точність (accuracy, ACR) | здатність ШІ забезпечувати близькість результатів виконання вимог та/або функцій, які представляються певними даними, до їх справжніх значень. | H | [2, 3, 23, 35] |
| 30 | Транспарентність (transparency, TRP) | здатність ШІ описувати, перевіряти та відтворювати моделі, окремі компоненти та алгоритми, за якими приймаються рішення. | H | [2, 3, 26, 27] |
| 31 | Функційна безпечність (safety, SFT) | здатність ШІ не припускати ризики неприйнятних пошкоджень і втрат внаслідок відмов, обумовлених внутрішніми і зовнішніми причинами, та мінімізувати їх наслідки з використанням засобів, вбудованих в ШІ. | H | [2, 3, 12] |
| 32 | Цілісність (integrity, ING) | здатність ШІ, яка характеризується ступенем запобігання несанкціонованому доступу задля модифікації алгоритмів або даних, використовуваних системою. | R | [21] |

## 1.4. Базова модель якості ШІ

### 1.4.1. Послідовність побудови моделі якості ШІ

При побудові ієрархії на цьому і подальших етапах використовуємо наступну процедуру:

Крок 1. Кожну з характеристик $S_{ChAI}$ співставляємо з усіма іншими і вибираємо такі, які залежать від інших, і є такими, від яких не залежать всі інші (відношення залежності визначається експертним шляхом). Такі характеристики мають бути віднесено до першого рівня ієрархії $S_{ChAI-1}$ (з потужністю $m_1$);

Крок 2. Характеристики, які не ввійшли до $S_{ChAI-1}$ , тобто сформували множину

$S_{ChAI-2} = S_{ChAI} \setminus S_{ChAI-1}$,

розділяються на $m_1$ підмножин $S_{ChAI-2i}$, які не перетинаються і впливають на відповідні характеристики з множини $S_{ChAI-1}$:

$S_{ChAI-2i} = \cup S_{ChAI-2i}$; $i = \{1,2,\ldots, m_1\}$,

$\forall$ $i, j = \{1,2,\ldots, m_1\}$, $i \neq j$: $S_{ChAI-2i} \cap S_{ChAI-2j} = ø$.

Крок 3. Операції 1,2 повторюються для кожної з підмножин $S_{ChAI-2i}$ потужністю $m_{2i}$, що надає змогу сформувати другий і третій рівень ієрархії.

Ця процедура продовжується далі, у разі більшої кількості півнів ієрархії. Моделі якості представлено у *графовій* формі, найбільш наочній та зручній для подальшого використання з метою оцінювання якості ШІ. В графі вершини відповідають характеристикам і підхарактеристикам, а ребра – відношенням залежності між ними.

Відповідно до покрокової процедури формуємо множину характеристик першого півня. До таких віднесено характеристики $S_{ChAI-1}$ = {ETH, EXP, LFL, RSP, TST}, оскільки вони є найбільш вживаними і впливають безпосередньо на якість ШІ.

Характеристиками другого рівня (підхарактеристиками) є такі:

- для ETH: $S_{ChAI-21}$ = {FRN, GRS, HMA, HMO, RDR};
- для EXP: $S_{ChAI-22}$ = {ACN, CSL, CMT, CMH, TRP, INP, INR, VFB};
- для LFL, RSP: $S_{ChAI-23} = S_{ChAI-24} = ø$;
- для TST: $S_{ChAI-25}$ = {DVS, RSL, RBS, SFT, SCR, ASP, ACR}.

Далі характеристиками третього рівня є $S_{ChAI-1}$ = {BIS, NDS, TRC, SWB, PRV, ING, OBC}.

Графова форма моделі надана на рис. 1.1.

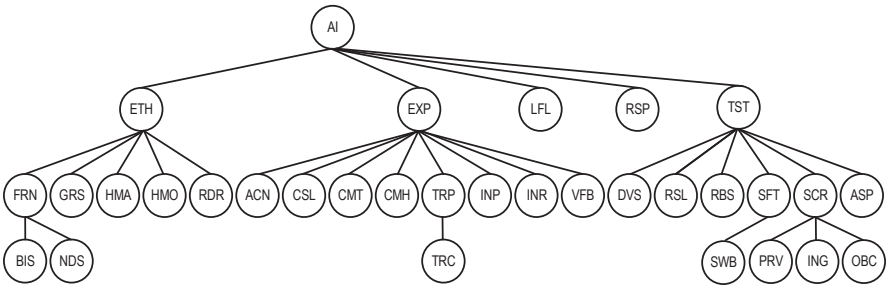



Рисунок 1.1 – Графова форма вихідної моделі якості штучного інтелекту

### 1.4.2. Особливості базової моделі якості ШІ

Базова модель якості ШІ розробляється для того, щоб зробити її більш компактною і зручною для інженерної практики для оцінювання реальних систем ШІ. Базова модель може бути отримана шляхом її оптимізації по «вертикалі» і «горизонталі» і відрізняється від загальної тим, що:

- оптимізація по вертикалі здійснюється шляхом представлення моделі двома рівнями. Підхарактеристики третього рівня ураховуються на рівні метрик відповідних характеристик другого рівня;
- відповідні складові характеристик, які видаляються або об’єднуються, можуть бути враховані на рівні метрик, що використовуються для оцінювання, та їх зважування відповідним чином при оцінюванні характеристики верхнього рівня;

- видалено характеристику RSP: вона перетинається з іншими характеристиками цього рівня:

а) довірчоздатністю TST – з точки зору відповідальності за виконання вимог користувача в цілому. Крім того, вимога щодо інформування у разі можливого їх порушення, яка є складовою відповідальності, може розглядатися як обов'язкова і враховуватися при оцінюванні довірчоздатності;

в) пояснюваністю EXP – з точки зору пристосованості до перевірки та надання інформації у разі порушення відповідних норм і вимог, що є складовими підхарактеристик TRP, VFB;

- характеристики HMA і HMO об'єднуються, оскільки вони, зазвичай, розглядаються разом і можуть доповнюватися на рівні метрик. Нове визначення HMA: здатність ШІ на підставі контролю надавати користувачу можливість приймати автономні обґрунтовані рішення щодо застосування і втручатися визначеним чином в функціонування ШІ;

- відстежуваність ACN і причинність CSL об'єднуються з TRP, оскільки можуть розглядатися як додаткові метрики транспарентності. Тоді транспарентність може визначатися як здатність ШІ описувати, перевіряти та відтворювати моделі, окремі компоненти та алгоритми, за якими приймаються рішення, визначати причино-наслідкові зв'язки між подіями і надавати звіти за визначеною формою про результати функціонування;

- характеристика ACP виключена як окрема, оскільки вона фактично є «м'якою» складовою власне TST, визначення якої не потребує коригування.

Базова модель описується графом (рис. 1.2), який є підграфом загальної моделі і містить 19 характеристик.

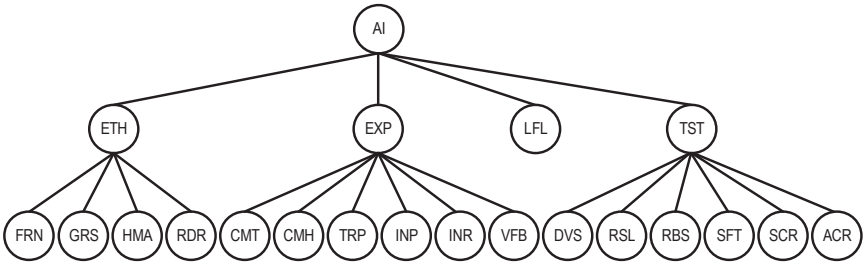



Рисунок 1.2 – Графова форма базової моделі якості штучного інтелекту

## 1.5. Приклад побудови моделей якості для систем штучного інтелекту

Розглянемо приклади побудови моделі якості для реальних систем штучного інтелекту на підставі запропонованої базової моделі.

Такі моделі можуть бути використано для обґрунтування вимог до розроблюваних систем або перевірки їх виконання і коригування проєктних рішень. Процес побудови моделей для реальних СШІ може називатися

розробленням профілю або профілюванням вимог. Профілювання реалізується шляхом визначення характеристик якості на кожному рівні ієрархії моделі, які є важливими для системи, що аналізується.

Ця задача розв'язується далі у експертний спосіб для двох систем з використанням базової моделі якості ШІ (рис. 1.2). Мета цієї частини дослідження – продемонструвати як можуть застосовуватися моделі на підставі запропонованих в підрозділі 1.4.

### 1.5.1. Система моніторингу інженерних комунікацій

Перший приклад стосується системи моніторингу інженерних комунікацій (СМІК), завдання якої полягає у розпізнаванні дефектів на стінках стічних труб [37]. У системі реалізується багатоетапний метод машинного навчання, перший етап якого полягає в контрастному самонавчанні на нерозмічених даних, а наступні етапи пов'язані з визначенням двійкового коду кожного класу, що використовується як мітка під час точного налаштування моделі. Модель якості СМІК як системи штучного інтелекту представлено на рис. 1.3.

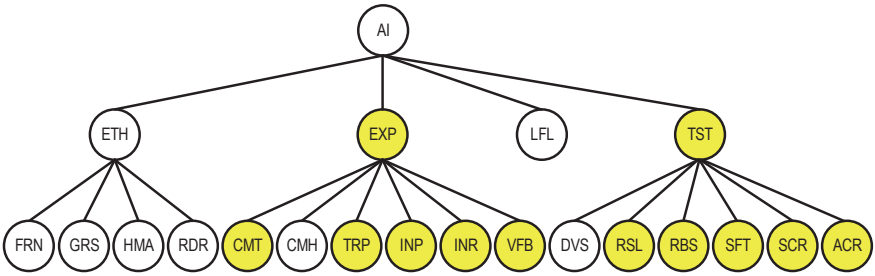



Рисунок 1.3 – Графова форма базової моделі якості системи моніторингу інженерних комунікацій

Її особливості є наступними:

- на першому рівні включено дві характеристики якості ШІ: пояснюваність EXP і довірчоздатність TST; етичність ETH і законність LFL виключені з розгляду, оскільки необхідність відповідати нормам моралі і права для системи не є природною;
- серед підхарактеристик для пояснюваності EXP включено всі підхарактеристики за виключенням зрозумілості CMH, враховуючі автономний режим роботи СМІК; для довірчоздатності TST також включено всі підхарактеристики за виключення диверсності DVS, оскільки застосування принципу багатоверсійності в СМІК обмежується необхідністю мінімізувати габаритно-масові та енергетичні показники.

1.5.2. Система розпізнавання дорожніх знаків

Другий кейс ілюструє побудову профіля якості для системи розпізнавання дорожніх знаків (СРДЗ), яка базується на технології розпізнавання зображень з використанням згорткових нейронних мереж [38]. У запропонованій системі вдосконалено етапи нормалізації та сегментації. На етапі нормалізації перед еквалізацією проводиться афінне перетворення зображення. Для сегментації та розпізнавання номерного знаку використовується нейронна мережа Mask R-CNN.

Модель якості СРДЗ як системи штучного інтелекту представлено на рис. 1.4.

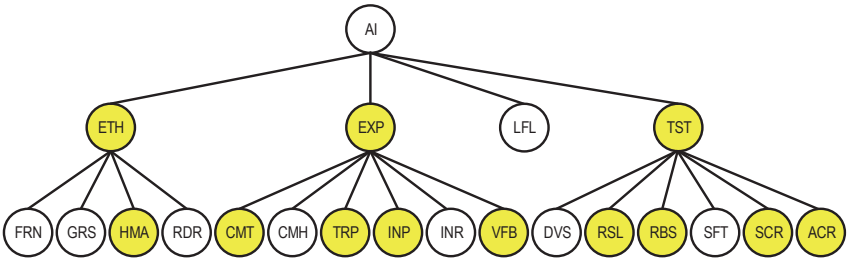



Рисунок 1.4 – Графова форма базової моделі якості системи розпізнавання дорожніх знаків

Її особливості є наступними:

- на першому рівні модель включає три характеристики (етичність ETH, пояснюваність EXP і довірчоздатність TST), законність LFL виключена з розгляду з причини відсутності нормативно-правових актів, які б регулювали правові аспекти застосування подібних систем;
- для етичності включена одна характеристика – людська автономність HMA, оскільки користувачу в ряді випадків може надаватися можливість остаточного прийняття рішень щодо результатів роботи СРДЗ;
- серед підхарактеристик для пояснюваності EXP включено всі підхарактеристики за виключенням зрозумілості CMH та інтерактивності INR, враховуючі автономний режим роботи СМІК;
- довірчоздатність представлена чотирма підхарактеристиками з шести за виключенням диверсності DVS і безпечності SFT, враховуючи відсутність функцій формування керуючих впливів на людину.

1.6. Висновки

Не зважаючи на велику кількість публікацій та документів, виданих поважними національними та міжнародними інституціями, на даний час

відсутня повна, упорядкована і несуперечлива сукупність характеристик ШІ, яку можна було б називати моделлю якості за аналогією з існуючими та загальноприйнятими моделями якості, розробленими для програмного забезпечення. Тому основним результатом дослідження вважаємо запропоновану модель якості штучного інтелекту, яка базується на аналізі та гармонізації визначень та залежностей характеристик якості, специфічних для ШІ.

Вибір характеристик та побудова моделі якості здійснювалось таким чином, щоб виключити повторення, забезпечити повноту представлення, а також визначити специфічні ознаки кожної з характеристик. Зрозуміло, що зробити модель, яка б повністю відповідала таким вимогам вкрай важко, тому представлені варіанти мають доповнюватися та удосконалюватися з урахуванням швидкого розвитку технологій і застосувань ШІ.

Основна та базова моделі якості надано в даному дослідженні у графовій формі, найбільш наочній та зручній для подальшого використання з метою оцінювання якості ШІ. Вони забезпечують можливість отримання часткових профілів якості з урахуванням специфіки відповідних систем, що було продемонстровано для двох СШІ. Вони далі можуть використовуватися як основа для метрично-базованого оцінювання якості таких систем.

Запропоновані моделі якості є відкритими і можуть доповнюватися і деталізуватися відповідно до специфіки призначення та сфери використання ШІ. На нашу думку, на базі запропонованих моделей можливе розроблення міжгалузевого стандарту якості та вимог до ШІ.

Подальші дослідження доцільно проводити за такими напрямами:

- профілювання (доповнення і деталізація) моделей для конкретних галузей, яке має супроводжуватися оглядом характеристик і підхарактеристик, що додаються на підставі досвіду розроблення та використання СШІ;
- розроблення метрик і алгоритмів для оцінювання ШІ за кожною з запропонованих характеристик та якості в цілому. Доцільно збирати та аналізувати інформацію про різні метрики задля їх включення до загальної бази даних;
- розроблення інструментальних засобів та кейс-орієнтованих методів оцінювання якості ШІ [39, 40]. Вони можуть базуватися на загальних Assurance Case підходах [41] і підходах, які стосуються оцінювання функційної і кібербезпеки [42]. Відбір інструментальних засобів, зокрема, для оцінювання кібербезпеки є окремою задачею, яка може виконуватися за допомогою засобів ШІ [43].

Робота підтримана проектом ECHO "European network of cybersecurity centres and competence hub for innovation and operations", який отримав фінансування від програми досліджень та інновацій Європейського Союзу Horizon 2020 в рамках грантової угоди № 830943. Автори вдячні колегам з консорціуму, співробітникам кафедри комп'ютерних систем, мереж і кібербезпеки Національного аерокосмічного університету ім. М. Є.

Жуковського «XAI» за участь в дискусіях, творчий аналіз результатів і цінні поради впродовж підготовки цієї статті.

Література

1. A Systematic Review of Explainable Artificial Intelligence in Terms of Different Application Domains and Tasks [Text] / M. R. Islam, M. U. Ahmed, S. Barua, S. Begum // Applied Sciences. – 2022. – Vol. 12. – Article Id: 1353. DOI: 10.3390/app12031353.

2. European Commission, High-Level Expert Group on Artificial Intelligence. Ethics Guidelines for Trustworthy AI [Electronic resource]. – Available at: https://www.aepd.es/sites/default/files/2019-12/ai-ethics-guidelines.pdf. – 10.03.2022.

3. European Commission, High-Level Expert Group on Artificial Intelligence. The Assessment List for Trustworthy Artificial Intelligence (ALTAI) [Electronic resource]. – Available at: https://airegio.ems-carsa.com/nfs/programme_5/call_3/call_preparation/ALTAI_final.pdf. – 10.03.2022.

4. ISO/IEC TR 24372:2021. Information technology. Artificial intelligence. Overview of computational approaches for AI systems [Electronic resource]. – Available at: https://www.iso.org/standard/78508.html. – 10.03.2022.

5. ISO/IEC TR 24028:2020. Information technology. Artificial intelligence. Overview of trustworthiness in artificial intelligence [Electronic resource]. – Available at: https://www.iso.org/standard/77608.html. – 10.03.2022.

6. ISO/IEC TR 24027:2021. Information technology. Artificial intelligence. Bias in AI systems and AI aided decision making [Electronic resource]. – Available at: https://www.iso.org/standard/77607.html. – 10.03.2022.

7. IEEE 2941-2021. Standard for Artificial Intelligence (AI) Model Representation, Compression, Distribution, and Management [Electronic resource]. – Available at: https://ieeexplore.ieee.org/document/6922153. – 10.03.2022.

8. Four Principles of Explainable Artificial Intelligence: Draft NISTIR 8312 / P. J. Phillips, C. A. Hahn, P. C. Fontana, D. A. Broniatowski, M. A. Przybocki,

C. A. Hahn, P. C. Fontana. – Gaithersburg: National Institute of Standards and Technology, 2020. – 24 p. DOI: 10.6028/NIST.IR.8312.

9. Towards a Standard for Identifying and Managing Bias in Artificial Intelligence: NIST Special Publication 1270 / R. Schwartz, L. Down, A. Jonas, E. Tabassi. – Gaithersburg: National Institute of Standards and Technology, 2021. – 77 p. DOI: 10.6028/NIST.SP.1270.

10. Trust and Artificial Intelligence: Draft NISTIR 8332 / B. Stanton, T. Jensen. – Gaithersburg: National Institute of Standards and Technology. – 2021. – 23 p. DOI: 10.6028/NIST.IR.8332-draft.

11. OECD. Tools for Trustworthy AI: A Framework to Compare Implementation Tools [Electronic resource]. – Available at:

https://www.oecd.org/science/tools-for-trustworthy-ai-008232ec-en.htm. – 10.03.2022.

12. UNESCO. Recommendation on the Ethics of Artificial Intelligence [Electronic resource]. – Available at: https://unesdoc.unesco.org/ark:/48223/pf0000381137. – 10.03.2022.

13. Christoforaki, M. AI Ethics–A Bird's Eye View / M. Christoforaki, O. Beyan // Applied Sciences. – 2022. – Vol. 12. – Article Id: 4130. DOI: 10.3390/app12094130.

14. Explainable AI: A Brief Survey on History, Research Areas, Approaches and Challenges [Text] / F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, J. Zhu // Natural Language Processing and Chinese Computing: collective monograph, edited by J. Tang, M. Y. Kan, D. Zhao, S. Li, H. Zan. – Berlin/Heidelberg: Springer International Publishing, 2019. – Vol. 11839. – P. 563-574. DOI: 10.1007/978-3-030-32236-6_51.

15. Trustworthy AI [Text] / R. Chatila, V. Dignum, M. Fisher, F. Giannotti, K. Morik, S. Russell, K. Yeung // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): collective monograph, edited by B. Braunschweig, M. Ghallab. – Cham: Springer International Publishing, 2021. – Vol. 12600. – P. 13-39. DOI: 10.1007/978-3-030-69128-8.

16. Gordieiev, O. IT-oriented software quality models and evolution of the prevailing characteristics [Text] / O. Gordieiev, V. Kharchenko // Dependable Systems, Services and Technologies (DESSERT): Proceeding of 9th Int. Conf., 2018. – P. 375-380. DOI: 10.1109/DESSERT.2018.8409162.

17. Gordieiev, O. Software quality standards and models evolution: greenness and reliability issues [Text] / O. Gordieiev, V. Kharchenko, M. Fusani // Information and communication technologies in education, research, and industrial applications: collective monograph, edited by V. Yakovyna, H. C. Mayr, M. Nikitchenko, G. Zholtkevych, A. Spivakovsky, S. Batsakis. – Berlin/Heidelberg: Springer International Publishing, 2016. – P. 38-55. DOI: 10.1007/978-3-319-30246-1_3.

18. Gerstlacher, J. Green and Sustainable Software in the Context of Software Quality Models [Text] / J. Gerstlacher, I. Groher, R. Plösch // HMD Praxis der Wirtschaftsinformatik. – 2021. – Article Id: 554. DOI: 10.1365/s40702-021-00821-0.

19. Software Quality for AI: Where We Are Now? [Text] / V. Lenarduzzi, F. Lomio, S. Moreschini, D. Taibi, D. A. Tamburri // Lecture Notes in Business Information Processing: collective monograph, edited by D. Winkler, S. Biffl, D. Mendez, M. Wimmer, J. Bergsmann. – Cham: Springer International Publishing, 2021. – Vol. 404. – P. 43-53. DOI: 10.1007/978-3-030-65854-0_4.

20. Smith, A. L. Quality characteristics of artificially intelligent systems [Text] / A. L. Smith, R. Clifford // CEUR Workshop Proceedings (CEUR-WS). – 2020. – Vol. 2800. – P. 1-6.

21. ISO/IEC 25010:2011. Systems and software engineering. Systems and software Quality Requirements and Evaluation (SQuaRE). System and software quality models [Electronic resource]. – Available at: https://www.iso.org/standard/35733.html. – 10.03.2022.

22. Gordieiev, O. Software individual requirement quality model [Text] / O. Gordieiev // Radioelectronic and Computer Systems. – 2020. – No. 94. – P. 48-58. DOI: 10.32620/reks.2020.2.04.

23. The Industrial Internet of Things. Trustworthiness Framework Foundations. An Industrial Internet Consortium Foundational Document. Version V1.00 – 2021-07-15 [Electronic resource]. – Available at: https://www.iiconsortium.org/pdf/Trustworthiness_Framework_Foundations.pdf. – 10.03.2022.

24. Cambridge Dictionary. Acceptability. [Electronic resource]. – Available at: https://dictionary.cambridge.org/dictionary/english/acceptability. – 10.03.2022.

25. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI [Text] / A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera // Information Fusion. – 2020. – Vol. 58. – P. 82-115. DOI: 10.1016/j.inffus.2019.12.012.

26. Adadi, A. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI) [Text] / A. Adadi, M. Berrada // IEEE Access. – 2018. – Vol. 6. – P. 52138-52160. DOI: 10.1109/ACCESS.2018.2870052.

27. Burciaga, A. Six Essential Elements of a Responsible AI Model [Electronic resource]. – Available at: https://www.forbes.com/sites/forbestechcouncil/2021/09/01/six-essential-elements-of-a-responsible-ai-model/?sh=21ebcb8456cf. – 10.03.2022.

28. Causability and Explainability of Artificial Intelligence in Medicine [Text] / A. Holzinger, G. Langs, H. Denk, K. Zatloukal, H. Müller // WIREs Data Mining and Knowledge Discovery. – 2019. – Vol. 9. – P. 1-13. DOI: 10.1002/widm.1312.

29. Cambridge Dictionary. Comprehensibility [Electronic resource]. – Available at: https://dictionary.cambridge.org/dictionary/english/comprehensibility. – 10.03.2022.

30. From “Explainable AI” to “Graspable AI” [Text] / M. Ghajargar, J. Bardzell, A. S. Renner, P. G. Krogh, K. Höök, D. Cuartielles, L. Boer, M. Wiberg // Tangible, Embedded, and Embodied Interaction (TEI) : Proceeding of 15th Int. Conf., 2021. – P. 1-4. DOI: 10.1145/3430524.3442704.

31. From Responsibility to Reason-Giving Explainable Artificial Intelligence [Text] / K. Baum, S. Mantel, E. Schmidt, T. Speith // Philosophy & Technology. – 2022. – Vol. 35. – Article Id: 12. DOI: 10.1007/s13347-022-00510-w.

32. Explaining explanations: An overview of interpretability of machine learning [Text] / L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, L. Kagal // Data Science and Advanced Analytics (DSAA) : Proceeding of 5th Int. Conf., 2018. – P. 80-89. DOI: 10.1109/DSAA.2018.00018.

33. Wright, D. Understanding "Trustworthy" AI: NIST Proposes Model to Measure and Enhance User Trust in AI Systems [Electronic resource] / D. Wright. – Available at: https://www.jdsupra.com/legalnews/understanding-trustworthy-ai-nist-6387341. – 10.03.2022.

34. Traceability for Trustworthy AI: A Review of Models and Tools [Text] / M. Mora-Cantallops, S. Sánchez-Alonso, E. García-Barriocanal, M.-A. Sicilia // Big Data and Cognitive Computing. – 2021. – Vol. 5, Iss. 2. – Article Id: 20. DOI: 10.3390/bdcc5020020.

35. When Autonomous Systems Meet Accuracy and Transferability through AI: A Survey [Text] / C. Zhang, J. Wang, G.G. Yen, C. Zhao, Q. Sun, Y. Tang, F. Qian, J. Kurths // Patterns. – 2020. – Vol. 1, Iss. 4. – P.1-28. DOI: 10.1016/j.patter.2020.100050.

36. Patil, K. R. Verifiability as a Complement to AI Explainability: A Conceptual Proposal [Preprint] [Electronic resource] / K. R. Patil, B. Heinrichs. – Available at: http://philsci-archive.pitt.edu/20297. – 10.03.2022.

37. Багатоетапний метод глибинного навчання з попереднім самонавчанням для класифікаційного аналізу дефектів стічних труб [Текст] / В. В. Москаленко, М. О. Зарецький, А. С. Москаленко, А. Г. Коробов, Я. Ю. Ковальський // Радіоелектронні і комп'ютерні системи. – 2021. – № 4. – С. 71-81. DOI: 10.32620/reks.2021.4.06.

38. Kuchuk, H. System of license plate recognition considering large camera shooting angles [Text] / H. Kuchuk, A. Podorozhniak, N. Liubchenko, D. Onischenko // Radioelectronic and Computer Systems. – 2021. – No. 4. – P. 82-91. DOI: 10.32620/reks.2021.4.07.

39. Felderer, M. Quality Assurance for AI-Based Systems: Overview and Challenges (Introduction to Interactive Session) [Text] / M. Felderer, R. Ramler // Lecture Notes in Business Information Processing: collective monograph, edited by D. Winkler, S. Biffl, D. Mendez, M. Wimmer, J. Bergsmann. – Cham: Springer International Publishing, 2021. – Vol. 404. – P. 33-42. DOI: 10.1007/978-3-030-65854-0_3.

40. Bloomfield, R. Security-informed safety: If it's not secure, it's not safe [Text] / R. Bloomfield, K. Netkachova, R. Stroud // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): collective monograph, edited by S. Tonetta, E. Schoitsch, F. Bitsch. – Cham: Springer International Publishing, 2013. – Vol. 8166. – P. 17-32. DOI: 10.1007/978-3-642-40894-6_2.

41. Potii, O. Advanced security assurance case based on ISO/IEC 15408 [Text] / O. Potii, O. Illiashenko, D. Komin // Advances in Intelligent Systems and Computing: collective monograph, edited by W. Zamojski, J. Mazurkiewicz, J. Sugier, T. Walkowiak, J. Kacprzyk. – Cham: Springer International Publishing, 2015. – Vol. 365. – P. 391- 401. DOI: 10.1007/978-3-319-19216-1_37.

42. Conception and application of dependable Internet of Things based systems [Text] / O. O. Illiashenko, M. A. Kolisnyk, A. E. Strielkina, I. V. Kotsiuba,

V. S. Kharchenko // Radio Electronics, Computer Science, Control. – 2020. – Vol. 4. – P. 139-150. DOI: 10.15588/1607-3274-2020-4-14.

43. Architecture and Model of Neural Network Based Service for Choice of the Penetration Testing Tools [Text] / A. G. Tetskyi, V. S. Kharchenko, D. D. Uzun, A. S. Nechausov //. International Journal of Computing. – 2021. – Vol. 20(4). – P. 513-518. DOI: 10.47839/ijc.20.4.2438.

# 2. МЕТОД І МОДЕЛІ МЕТРИЧНОГО ОЦІНЮВАННЯ ЯКОСТІ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

**І. О. Васильєв[1], С. І. Доценко[2], О. І. Морозова[1], В. С. Харченко[1]**

*[1]Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»*
*[2]Український державний університет залізничного транспорту*

## 2.1. Вступ

Мотивація. В останній час розробляються і впроваджуються різні засоби штучного інтелекту (ШІ), що виконують відносно прості, на перший погляд, завдання, які займали багато часу та зусиль у минулому, зокрема, оброблення багаторозмірних зображень та відео, анімації статичних зображень, розпізнавання облич, розроблення чат-ботів та інші. Активно досліджуються і створюються засоби ШІ для більш складних задач, а саме, встановлення хвороби пацієнта на основі симптомів, розроблення асистентів для пілотів літака, виявлення зловмисників тощо [1, 2].

Вкрай важливо розуміти, чи можна довіряти системам, що базуються на використанні штучного інтелекту (СШІ). Велика кількість сучасних СШІ побудовано за принципом «чорної скриньки», тобто незрозуміло, яким чином вони працюють, а тільки є результати роботи. Важко перевірити, яким чином ШІ приймає рішення, чи є вони взагалі вірним або помилковим. Також потрібні засоби для порівняння декількох СШІ. У випадку, коли декілька варіантів ШІ конкурують щодо використання у деякій системі, потрібно визначити кращий.

Аналіз публікацій. Модель якості СШІ може бути представлено у вигляді графу типу «дерево», що надає упорядковану ієрархію характеристик [3]. Вона будується за аналогією з моделями якості програмного забезпечення [4-6]. Характеристики відібрано на підставі аналізу документів [6-10], гармонізації їх визначень та пошуку залежностей відповідно до [3].

Ці складові можуть далі розділятися на свої субхарактеристики. У кожної складної характеристики повинно буди щонайменше дві субхарактеристики. Отже, використані в даному досліджені моделі базуються на результатах роботи [3], яка була апробована для побудови моделей якості СШІ, описаних в [11, 12].

Метою дослідження є розроблення модель-базованого фреймворку для оцінювання якості СШІ з використанням метрик і методу візуалізації результатів. Відповідно до мети в статті аналізуються моделі якості СШІ (підрозділ 2.2), метрики і види згорток для її оцінювання (підрозділ 2.3), пропонується метод оцінювання якості та візуалізації результатів (підрозділ 2.4) і приклад використання методу (розділ 2.5). У висновках аналізуються основні результати дослідження та розроблення відповідного

інструментального засобу, а також описуються наступні кроки, спрямовані на розвиток моделей і методів для різних сфер застосування СШІ.

## 2.2. Моделі якості СШІ

**Загальна структура моделі.** В основі моделі (рис. 2.1) лежить оцінка загальної якості системи (artificial intelligence system, AIS).

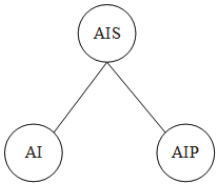



Рисунок 2.1 – Модель якості СШІ

Вона розділяється на два складових [3]:

– платформа ШІ (artificial intelligence platform, AIP) – середовище, у якому працює штучний інтелект. Вона відповідає за взаємодію з користувачем, управління штучним інтелектом, передачу йому даних та оброблення результатів. Платформа являє собою звичайну програму, написану людиною. Це може бути додаток на ПК, або хмарний сервіс. Через це, загалом, вимоги до платформи співпадають з вимогами до звичайних програм;

– модель ШІ (artificial intelligence, AI) – навчений штучний інтелект, шо приймає деякі дані на вході, да формує результати обчислень або керуючі впливи (прийняті рішення) на виході.

**Модель якості платформи ШІ** (рис. 2.2) включає такі характеристики [3,6,7]:

– доступність (accessibility, ACS);

– точність (accuracy, ACR);

– аудитопридатність (auditability, ADT);

– готовність (availability, AVL);

– керованість (controllability, CNT) – що включає керування даними (data governance, DGV) та керування функціями (function governance, FGV);

– ефективність (effectiveness, EFC);

– інформативність (informativeness, INF);

– надійність (reliability, RLB);

– обслуговуваність (maintainability, MNT) – що включає переносимість (transferability, TRF), ремонтопридатність (repairability, RPR) та модифікованість (modifiability, MDF);

– сталість (sustainability, SST);

– зручність (usability, USB).

**Модель якості ШІ** (рис. 2.3) включає п’ять важливих характеристик [3,6,7]:

– етичність (ethics, ETH) – здатність ШІ відповідати діючим нормам моралі за результатами функціонування. Вона має субхарактеристики: справедливість (fairness, FRN), сприйнятливість (graspability, GRS), людська автономність (human agency, HMA), людський нагляд (human oversight, HMO) та відшкодовуваність (redress, RDR);

– пояснюваність (explainability, EXP) – здатність ШІ бути зрозумілим і передбачуваним з точки зору призначення та поведінки. Вона включає субхарактеристики: відстежуваність (accountability, ACN), причинність (causality, CSL), завершеність (completeness, CMT), зрозумілість (comprehensibility, CMH), прозорість (transparency, TRP), простежуваність (traceability, TRC), описуваність (descriptiveness, DSC), інтерпретабельність (interpretability, INP), інтерактивність (interactivity, INR), верифікованість (verifiability, VFB);

– законність (lawfulness, LFL) – здатність ШІ відповідати законодавчим і нормативно-правовим актам;

– відповідальність (responsibility, RSP) – здатність ШІ функціонувати з урахуванням очікувань замовника (користувача) у відповідності до етичних норм, законодавчих нормативно-правових актів, а також інформувати його при їх порушення;

– довірчоздатність (trustworthiness, TST) - здатність ШІ, яка характеризується ступенем впевненості користувача або іншої зацікавленої особи (розробника, аудитора тощо) в тому, що ШІ відповідає вимогам і виконує функції у передбачуваний спосіб. Вона складається з субхарактеристик: диверсність (diversity, DVS), резильєнтність (resiliency, RSL), робастність (Robustness, RBS), функційна безпечність (safety, SFT), захищеність (інформаційна/кібербезпека) (security, SCR), приватність (privacy, PRV), цілісність (integrity, ING), об’єктивність (objectivity, OBC), прийнятність (acceptability, ACP).

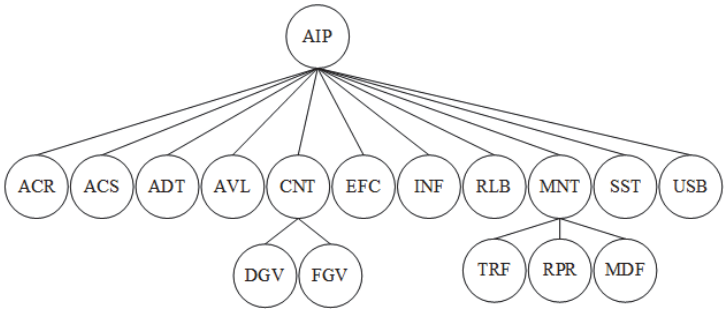



Рисунок 2.2 – Платформа ШІ

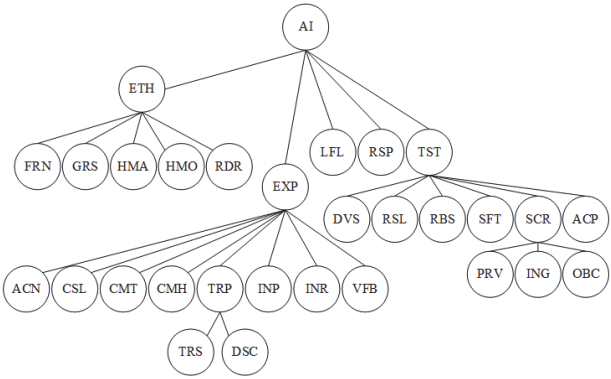



Рисунок 2.3 – Модель ШІ

2.3. Метрики для оцінювання якості СШІ

**Метрики якості.** Для розрахунку значень характеристик у моделі використовуються різні метрики. Кожна характеристика нижнього рівня повинна мати хоча б одну метрику. Ці метрики встановлюються в залежності від конкретного типу СШІ, вимог до системи, особливостей її розробки тощо. Результатами розрахунку метрик є показник характеристики, який може бути [5]:

– метричний (абсолютний або відносний);

– порядковий (рівневий);

– номінальний (виконується повністю, частково або зовсім не виконується).

Оскільки декілька показників можуть мати різні типи (кількісні та якісні), необхідний спосіб переведення одних показників в інші. Для цього їх потрібно нормалізувати та привести до єдиної шкали від 0 до 1. Для нормалізації метричних показників можна використовувати формулу 2.1.

$$p = \frac{m - m_{min}}{m_{max} - m_{min}}, \tag{1}$$

де $p$ – нормалізоване значення показника; $m$ – оцінене значення показника; $m_{max}$ – максимальне значення показника (або значення при якому вважається що система повністю відповідає характеристиці); $m_{min}$ – мінімальне значення показника (або значення, при якому вважається, що система повністю не відповідає вимогам до характеристики).

Для якісних показників кожне з можливих значень шкали відповідає певному метричному значенню, граничні значення повинні дорівнювати 0 і 1.

Розподілення між шкалою необов'язково має бути пропорційним. Приклад нормалізації якісної шкали зображено в табл. 2.1.

Таблиця 2.1 – Нормалізація порядкової шкали

| Вихідна порядкова шкала | Нормалізоване значення |
|---|---|
| Не відповідає | 0 |
| Відповідає частково | 0,33 |
| Відповідає повністю | 1 |

Оскільки одні характеристики можуть бути більш важливими, ніж інші, вводяться вагові коефіцієнти, які змінюються від 0 до 1.

Вони визначають, наскільки ця субхарактеристика є важливою та яку частину вона складає у характеристиці вищого рівня. Сума вагових коефіцієнтів субхарактеристик однієї характеристики завжди має дорівнювати одиниці.

**Згортки.** Для характеристик проміжних рівнів та якості в цілому виконується операція згортки. Вона полягає у розрахунку якості характеристики на основі значень метрик її субхарактеристик. Ця операція може мати декілька варіантів реалізації, як це визначено в [5] для оцінки якості ПЗ:

– адитивна згортка;
– згортка на основі нечітких операцій;
– предикативна згортка;
– булева згортка;
– комбінована згортка.

Адитивна згортка полягає у сумі зважених нормалізованих показників усіх субхарактеристик (2.2).

$$P = \sum_{i=1}^{n} w_i p_i, \tag{2.2}$$

де $P$ – значення показника характеристики; n – кількість субхарактеристик характеристики; $w_i$ – ваговий коефіцієнт і-ої субхарактеристики; $p_i$ – значення показника і-ої субхарактеристики.

Згортка на основі нечітких операцій полягає у формуванні правил розрахунку значення показника характеристики. Такі правила можуть бути будь-якими, головне, щоб у результати був отриманий нормалізований показник.

У предикативній згортці для пошуку значення показника формується набір предикатів – логічних правил, за якими він визначається.

У булевій згортці значення усіх показників мають набувати булевих значень, тобто 0 або 1. Значення показника характеристики виконується за певною булевою функцією, та набуває значення 0 – не відповідає, або 1 – відповідає.

У комбінованій згортці використовуються декілька згаданих методів – це необхідно, якщо у субхарактеристиках задіяні різні види показників, як порядкових, так і метричних. Тоді має виконуватися окрема згортка показників кожного типу, щоб отримати кінцеве значення інтегрованого показника.

2.4. Метод оцінювання та візуалізації результатів

Радіальні метричні діаграми. Для візуалізації моделі використовуються радіальні метричні діаграми (РМД) [5]. Загальна схема такої діаграми зображена на рис. 2.4.

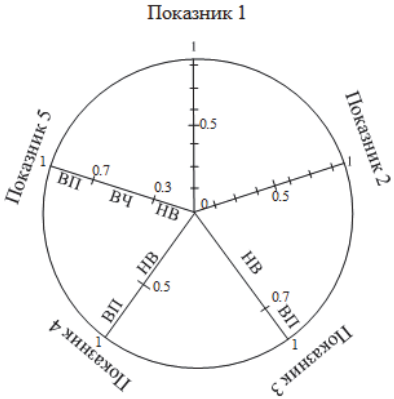



Рисунок 2.4 – Схема РМД

Центральна точка діаграми є точкою відліку показників. Для кожної з субхарактеристик проводиться вісь з цієї точки. На цьому промені встановлюється шкала оцінювання, в залежності від типу показника. На рис. 2.4 показники 1 та 2 показують приклад метричних показників, 3 та 4 – приклад двозначного показника («відповідає», «не відповідає»), 5 – тризначного показника («відповідає», «відповідає частково», «не відповідає»).

Під час оцінки, на шкалі позначається поточне значення показника. Потім усі сусідні позначені точки об'єднуються лініями та отримана фігура замальовується. Така діаграма дозволяє наочно показати рівень оцінки характеристики. Чим більше площа отриманої фігури, тим вище рівень якості. Також одразу видно, які з субхарактеристик не відповідають потребам якості.

Послідовність оцінювання. Оцінювання за допомогою моделі якості виконується у наступні кроки:

1) встановити характеристики, які будуть оцінюватися, з урахуванням стандартів галузі роботи СШІ, вимог до системи тощо;

2) визначити ступінь впливу цих характеристик на якість системи, та встановити вагові коефіцієнти для усіх характеристик;

3) встановити метрики для оцінювання усіх характеристик нижніх рівнів, та задати їх мінімальне та максимальне значення, за якими вони набувають значень 0 та 1 відповідно;

4) обрати тип згортки для розрахунку значень характеристик вищих рівнів;

5) розрахувати значення показників нижчих рівнів за встановленими метриками;

6) за встановленими методами згортки, розрахувати інші характеристики;

7) візуалізувати необхідні характеристики за допомогою РМД.

## 2.5. Приклад оцінювання якості СШІ

При оцінюванні моделі ШІ для автопілоту автомобіля [13] були відібрані характеристики, для яких відповідні вершини маркуються сірим кольором на рисунку 2.5. Для оцінювання довірчоздатності експертним шляхом встановлено вагові коефіцієнти (табл. 2.2).

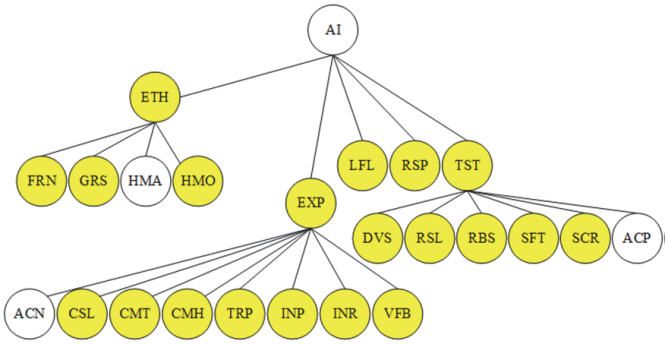



Рисунок 2.5 – Модель ШІ автопілоту

Таблиця 2.2 – Розрахунок оцінки довірчоздатності

| Довірчоздатність | | |
|---|---|---|
| Субхарактеристика | Ваговий коефіцієнт | Значення показника |
| Диверсність | 0,1 | 0,7 |
| Резильєнтність | 0,2 | 0,9 |
| Робастність | 0,2 | 0,9 |
| Функційна безпечність | 0,3 | 0,95 |
| Інформаційна захищеність | 0,2 | 0,8 |
| Значення довірчоздатності | | 0,875 |

Потім визначено як приклад фіксовані значення метрик субхарактеристик. Ці значення можуть бути обраховано шляхом обчислення відношення кількості успішних тестів (або кількості вимог, які виконано) для відповідної субхарактеристики до їх загальної кількості. Тести і вимоги можуть, залежно від їх важливості, також бути зваженими, що підвищує точність метрик.

За допомогою адитивної згортки (2) розраховується показник довірчоздатності. На цій підставі розробляється РМД, для відображення рівня довірчоздатності СШІ (рис. 2.6). Її значення визначає відповідну складову для РМД якості ШІ в цілому (рис. 2.7).

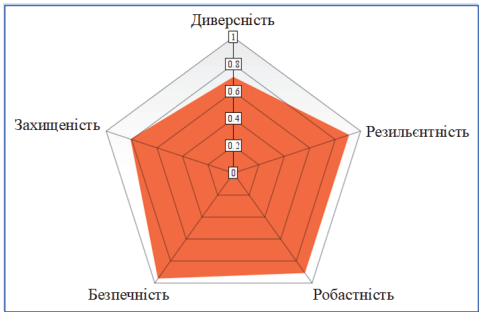



Рисунок 2.6 – РМД довірчоздатності

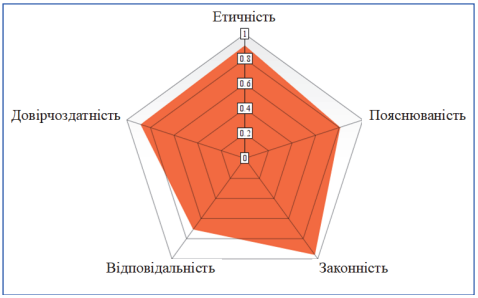



Рисунок 2.7 – РМД моделі ШІ

## 2.6. Висновки

Результати. Для оцінювання СШІ використано базові моделі якості, запропоновані в [3] і об’єднані у чотирьохрівневу ієрархію.

Для цих характеристик визначено правила формування метрик і метод розрахунку якості з використанням згорток та візуалізації проміжних і кінцевих результатів за допомогою радіальних метричних діаграм.

Відповідні моделі якості, метрики і методи оцінювання і візуалізації утворюють фреймворк для автоматизації процесів, який реалізується з використанням розробленого інструментального засобу.

Цей засіб дозволяє користувачу створювати модель якості СШІ (або використовувати запропоновану в [3] і адаптовану у цій статті), встановлювати метрики якості, вводити значення показників метрик. Потім на основі цих показників розраховується узагальнена метрика якості системи та візуалізується за допомогою РМД. Засіб є десктопним застосунком, створеним на платформі .Net Framework.

Майбутні кроки можуть бути присвячено розвитку моделі та інструментарію (метрик, методик і засобів) оцінювання якості для різних доменів (оборона, медицина, юриспруденція, інтерактивне мистецтво тощо) з урахуванням аспектів еволюції якості [14].

Література

1. Trustworthy AI [Text] / R. Chatila, V. Dignum, M. Fisher, F. Giannotti, K. Morik, S. Russell, K. Yeung // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): collective monograph, edited by B. Braunschweig, M. Ghallab. – Cham: Springer International Publishing, 2021. – Vol. 12600. – P. 13-39. DOI: 10.1007/978-3-030-69128-8.

2. A Systematic Review of Explainable Artificial Intelligence in Terms of Different Application Domains and Tasks [Text] / M. R. Islam, M. U. Ahmed, S. Barua, S. Begum // Applied Sciences. – 2022. – Vol. 12. – Article Id: 1353. DOI: 10.3390/app12031353

3. Харченко В. С., Фесенко Г. В., Ілляшенко О. О. (2022), Базова модель нефункційних характеристик для оцінки якості штучного інтелекту // Радіоелектронні і комп'ютерні системи 2(102). с. 1-14.

4. ISO/IEC 25010 (2011). ISO/IEC 25010:2011, Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.

5. Харченко В.С., Жихарев В.Я., Іллюшко В.М. та ін. (2004), Основи надійності цифрових систем, Харків: Нац. Аерокосм. Ун-т. «ХАІ».

6. NIST Four Principles of Explainable Artificial Intelligence: Draft NISTIR 8312 / P. J. Phillips, C. A. Hahn, P. C. Fontana, D. A. Broniatowski, M. A. Przybocki, C. A. Hahn, P. C. Fontana. – Gaithersburg: National Institute of Standards and Technology, 2020. – 24 p. DOI: 10.6028/NIST.IR.8312.

7. European Commission, Directorate-General for Communications Networks, Content and Technology, Ethics guidelines for trustworthy AI, Publications Office, (2019), https://data.europa.eu/doi/10.2759/346720

8. UNESCO (2021), Recommendation on the Ethics of Artificial Intelligence, https://unesdoc.unesco.org/ark:/48223/pf0000380455, Дата звернення 21.05.2022

9. ISO/IEC TR 24028:2020. Information technology. Artificial intelligence. Overview of trustworthiness in artificial intelligence [Electronic resource]. – Available at: https://www.iso.org/standard/77608.html. – 10.03.2022.

10. OECD. Tools for Trustworthy AI: A Framework to Compare Implementation Tools [Electronic resource]. – Available at: https://www.oecd.org/science/tools-for-trustworthy-ai-008232ec-en.htm. – 10.03.2022.

11. Москаленко В. В. Багатоетапний метод глибинного навчання з попереднім самонавчанням для класифікаційного аналізу дефектів стічних труб [Текст] / В. В. Москаленко, М. О. Зарецький, А. С. Москаленко, А. Г. Коробов,

Я. Ю. Ковальський // Радіоелектронні і комп'ютерні системи. – 2021. – № 4. – С. 71-81. DOI: 10.32620/reks.2021.4.06.

12. Kuchuk, H. System of license plate recognition considering large camera shooting angles [Text] / H. Kuchuk, A. Podorozhniak, N. Liubchenko, D. Onischenko // Radioelectronic and Computer Systems. – 2021. – No. 4. – P. 82-91. DOI: 10.32620/reks.2021.4.07

13. Some, Evariste & Gondwe, Greg & Rowe, Evan. (2019). Cybersecurity and Driverless Cars: In Search for a Normative Way of Safety. 352-357. 10.1109/IOTSMS48152.2019.8939168.

14. Gordieiev, O. IT-oriented software quality models and evolution of the prevailing characteristics [Text] / O. Gordieiev, V. Kharchenko // Dependable Systems, Services and Technologies (DESSERT): Proceeding of 9th Int. Conf., 2018. – P. 375-380. DOI: 10.1109/DESSERT.2018.8409162.

# 3. МЕТОДИ ОЦІНЮВАННЯ ПОЯСНЮВАНОГО ШТУЧНОГО ІНТЕЛЕКТУ ЯК СЕРВІСУ

**О. Ю. Веприцька, В. С. Харченко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 3.1. Вступ

Науковий прогрес на даному етапі розвитку технологій часом неможливо відрізнити від магії. Роботи здатні аналізувати навколишнє середовище, розумні системи можуть розпізнавати обличчя, безпілотні автомобілі та багато інших засобів вже впроваджено в повсякденне життя. З кожним днем людство все більше й більше винаходить способи автоматизації буденних рутинних справ та створює системи здатні виконати майже будь-яку роботу. Може виникнути питання чому з'являється необхідність заміняти людей? - роботи / комп'ютерні програми / машини набагато швидші, ефективніші, дешевші, а в деяких випадках навіть безпечніші за людей. Більшою мірою так стається через відсутність у них людських потреб та здатність обробляти велику кількість даних.

Переважна більшість подібних "загадкових" інструментів сучасності базуються на штучному інтелекті. Згідно з визначенням[1] штучний інтелект - це здатність комп'ютера навчатися та приймати рішення, як це може людина та інші види тварин.

Згідно з показниками Google Trends популярність сфери дослідження AI стрімко зростає в останні роки (рис. 3.1).


Рисунок 3.1 – Графік зацікавленості людства у сфері AI за останні 5 років.

Популярність та розповсюдження AI рішень обумовлює впровадження AI в найрізноманітніші домени [2]: Фінанси; Індустрія подорожей; Здоров'я; Транспортування; Роздрібна торгівля; Журналістика; Освіта; Сільське господарство; Розваги. Наприклад, програма грантів Nesta's Centre for Collective Intelligence Design надала фінансування різним командам у всьому світі, які розробили експерименти, щоб досліджувати та перевіряти цю ідею колективного інтелекту новими способами. В цьому випадку колективний

інтелект - людський інтелект у поєднанні з інтелектом машини. Попри складності, у знаходженні "мосту" між дата саєнтістами та бізнес-аналітиками та проблеми з якістю даних через відсутність єдиного джерела правди, співпраця компаній з різних сфер можлива та вже покращує деякі процеси: Wipro об'єдналася з компанією Transcell Oncologic, що займається технологіями стовбурових клітин, щодо використання її платформи штучного інтелекту Holmes для підвищення безпеки вакцин або ж Австралійський Університет Мельбурна об'єднався з низкою організацій державного та приватного секторів, щоб створити програму штучного інтелекту, яка може передбачати затори на дорозі до трьох годин наперед, а також оптимізувати рух і підвищити безпеку на дорозі[3]. В перспективі подібні об'єднання траплятимуться частіше.

Незважаючи на переваги що дають AI системи, сучасна технологія штучного інтелекту все ще далека від ідеалу. Більшою мірою завдяки неможливості передбачити/попередити критичні помилки, які можуть спричинити катастрофу без можливості подальшого відновлення. Найяскравішими прикладами критичних помилок AI є [4]:

- Microsoft Tay Chatbot - чат-бот, розроблений Microsoft, у своєму обліковому записі в Twitter почав лаятись, роблячи расистські зауваження та провокаційні політичні заяви. Microsoft довелося вимкнути бота менш ніж через 24 години після його запуску;
- Amazon's Recruiting Tool - програмне забезпечення Amazon для наймання на роботу зазнало невдачі, оскільки в ньому виникли гендерні упередження;
- Inverness Caledonian Thistle F.C. Ball Tracking System - система неодноразово плутала м'яч із лисою головою, особливо коли м'яч знаходився в нечітких областях (тобто був заблокований гравцями або коли він перебував у тіні, створеній стадіоном);
- Uber Self Driving Car - пішохід був збитий прототипом самокерованого автомобіля Uber через недостатню можливість штучного інтелекту «класифікувати об'єкт як пішохода, якщо цей об'єкт не знаходиться поблизу пішохідного переходу»;
- Face ID Hacked Using a 3D Printed Mask - розпізнавання обличчя було обмануто за допомогою надрукованої на 3D маски, яка зображує обличчя, яке використовується для автентифікації системи Facial ID.

Подібні випадки змушують людей сумніватися безпечності використання AI систем та й у самій концепції AI. Основними факторами неправильної роботи системи є недостатнє забезпечення ключовими характеристиками AI, описаними в[5], якістю тренувальних даних та недосконалим методам тестування й верифікації. Також складністю є факт що навіть після збою системи проблематично знайти джерело або причини помилки.

Хоч те що AI набирає популярність у багатьох сферах, розробка AI з нуля та її впровадження пов'язано з труднощами:

- вартість для малого та середнього бізнесу занадто висока;
- відсутність у підприємців спеціалістів з відповідними технічними знаннями, що можуть керувати розробкою, розгортанням та моніторингом AI систем;
- збір даних для навчання та їхнє зберігання потребує багато ресурсів, та часу для аналізу якості даних;
- етичні проблеми та непорозуміння при неправильно натренованій моделі AI, що приймає якісь рішення з упередженням до деяких категорій даних;
- високі вимоги до якості обладнання.

Отже, попит на готові рішення "з коробки" зростає. У зв'язку з цим наразі відбувається вивчення/дослідження/розробка eXplainable AI та Artificial Intelligence as a Service (AIaaS) (рис. 3.2). Дані отримані на основі показника об'єму пошуку (Google Trends) включаючи усі можливі регіони.

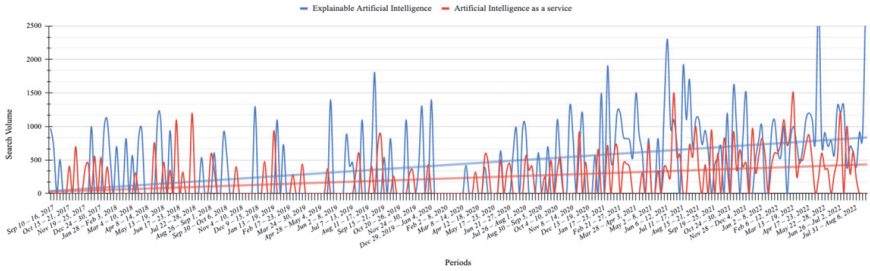



Рисунок 3.2 – Графік зацікавленості людства у сфері eXplainable AI та AIaaS за останні 5 років

## 3.2 Аналіз можливостей штучного інтелекту як сервісу AiaaS

Комерційні готові рішення COTS являють собою вже відпрацьовані програмні та апаратні продукти, доступні для використання замовниками та інтеграторами. Безумовною перевагою продуктів AI off the Shelf (AIOTS) є відносно низька вартість, а також некритичність щодо детального знання машинного навчання користувачами. Однією з концепцій COTS є послуга Everything as a Service (EaaS або XaaS). Сервіс XaaS охоплює чималий спектр інструментів і програм, які фізично розміщуються в хмарних середовищах і доступні звідусіль. Найпоширенішими є наступні: Software as a Service (SaaS); Platform as a Service (PaaS); Infrastructure as a Service (IaaS); Storage as a Service (StaaS); Disaster Recovery as a Service (DRaaS); Communications as a Service (CaaS); Network as a Service (NaaS); Testing as a Service (TaaS). Перевагами даного підходу є: фінансова складова; зменшення складності розробки додатків; масштабованість та доступність. Зацікавленість бізнесу в сервісах сприяє активному розвитку та просуванню AIaaS. Як зазначено в [6], AIaaS

поєднує ШІ з хмарними обчисленнями. Метою AIaaS є зробити ШІ доступним для всіх, незалежно від розміру організації, її технологічного розвитку або бюджету. В табл. 1 надано результати аналізу існуючих AIaaS послуг провідних провайдерів Amazon Web Services (AWS), Google Cloud Platform та Microsoft Azure.

Таблиця 3.1 – Artificial intelligence (AI) сервіси, що надаються хмарними провайдерами, та їхнє призначення

| **Тип AI** | **Azure** | **AWS** | **Google Cloud** |
|---|---|---|---|
| **Мовний** | *Автоматично перетворення аудіомовлення в текст* | | |
| | Speech to text | Amazon Transcribe | Speech-to-Text |
| | *Перетворення тексту у реальну мову* | | |
| | Text to speech | Amazon Polly | Text-to-Speech |
| | *Динамічний машинний переклад* | | |
| | Speech translation | Amazon Translate | Translation AI |
| | *Перевірка та ідентифікація мовців за їх унікальними голосовими характеристиками* | | |
| | Speaker recognition | | |
| **Візуальний** | *Автоматичне вилучення даних тексту з документів* | | |
| | Azure Form Recognizer | Amazon Textract | Document AI |
| | *Комп'ютерний зір* | | |
| | Computer Vision | Amazon Rekognition | Vision AI |
| | Face API | | |
| | Custom Vision | | Video AI |

Продовження табл. 3.1

| | | | |
|---|---|---|---|
| | *Виявлення аномальних дій та даних* | | |
| | Anomaly Detector | Amazon Fraud Detector | |
| | *Персоналізовані рекомендації продуктів* | | |
| | Personalizer | Amazon Personalize | Recommendations AI |
| | *Інше (аналіз бізнес-метрик, вразливостей коду, потенційно небажаних даних)* | | |
| **Аналітичний** | Content Moderator | Amazon Forecast<br>Amazon CodeGuru | |
| | *Чат-боти та обробка природної мови* | | |
| | Azure Bot Service | Amazon Lex | Dialogflow |
| | Cognitive Service for Language | Amazon Comprehend | Natural Language AI |
| | *Високоточний пошук* | | |
| **Інтерактивний** | Azure Cognitive Search | Amazon Kendra | |

Деякі сервіси наділені здібностями різних типів ШІ, але, враховуючи кінцевий результат, їх поділено на чотири типи: мовний, візуальний, аналітичний та інтерактивний. Мовний ШІ переважно використовується для розпізнавання тексту та його конвертацію в мовлення, візуальний ШІ – для розпізнавання об'єктів на зображеннях і відео, аналітичний ШІ – для вилучення, структурування, оброблення, пошуку шаблонів та залежностей у даних. Інтерактивний ШІ розрахований на пряму взаємодію з користувачем для оперативного розуміння та оброблення його запитів. Деякі послуги, схожі за функціями, надаються усіма компаніями, але є унікальними, а саме розпізнавання голосу, виявлення вразливостей коду та аналіз бізнес-метрик.

## 3.3. Аналіз вимог до AIaaS

### 3.3.1. Вимоги як до хмарних сервісів за стандартом IEC25010

Для AIaaS, залучених у критичних галузях, важливим є дотримання вимог як функціональних, так і нефункціональних. Оскільки сервіс AIaaS є підтипом XaaS, який є програмним забезпеченням (ПЗ), AIaaS має ієрархічно наслідувати вимоги до ПЗ та XaaS (рис. 1).

Не функційними вимогами ПЗ за стандартом IEC25010 [7] є наступні:

- функціональна придатність – ступінь, до якої продукт або система забезпечує функції, які відповідають заявленим потребам при використанні за певних умов;
- ефективність виконання – продуктивність роботи щодо кількості ресурсів, які використовуються за вказаних умов;
- сумісність – ступінь, до якої продукт, система чи компонент можуть обмінюватися інформацією з іншими продуктами, системами чи компонентами та (або) виконувати свої функції під час спільного використання одного апаратного чи програмного середовища;
- зручність використання – ступінь, до якої продукт або систему можуть використовувати певні користувачі для досягнення визначених цілей з ефективністю та задоволенням у визначеному контексті;
- надійність – ступінь, до якого система, продукт або компонент виконують задані функції за певних умов протягом певного проміжку часу (ця характеристика має підхарактеристики – зрілість); доступність; відмовостійкість; відновлюваність;
- безпека – ступінь, за яким продукт або система захищає інформацію та дані, щоб люди або інші продукти чи системи мали доступ до даних, відповідний їхнім типам і рівням авторизації;
- можливість підтримки – ступінь ефективності, з якою продукт або систему можна модифікувати, щоб покращити, виправити або адаптувати до змін навколишнього середовища та вимог;
- портативність – ступінь ефективності, з якою система, продукт або компонент можуть бути перенесені з одного апаратного, програмного або іншого операційного середовища в інше.

З огляду на те, які переваги повинні надаватись XaaS, виділимо наступні не функційні вимоги [8]:

- швидка реалізація – можливість швидкого підключення (інтеграції) готових рішень та сервісів до чинних систем;
- модифікація – безперервний процес покращення сервісу для збільшення ефективності та відповідності сучасним трендам;

- заощадження коштів – готове рішення вже розроблено і розгорнуто, що зменшує витрати на написання з нуля та процес розгортання, керування ресурсами;
- гнучкість – сервіс має надавати можливість кастомізації компонентів під кожного, чи перелік параметрів, за якими можна налаштовувати сервіс.

### 3.3.2. Вимоги і характеристики штучного інтелекту

Системи ШІ мають відповідати специфічним вимогам за результатами аналізу стандартів, керівних документів і наукових досліджень. Ці вимоги стосуються складових якості ШІ, платформ і систем ШІ, які можуть бути подані у вигляді моделей якості, описаних в [9]. На першому рівні ієрархії однієї з базових моделей якості ШІ розміщені чотири характеристики (див. рис. 3.3):

- етичність (ETH) – здатність ШІ відповідати чинним нормам моралі за результатами функціонування; її підхарактеристиками є справедливість (FRN), сприйнятливість (GRS), людська автономність (HMA), відшкодовуваність (RDR);
- пояснюваність (EXP) – здатність ШІ бути зрозумілим і передбачуваним щодо призначення та поведінки; складається з таких підхарактеристик: завершеності (CMT), транспарентності (TRP), інтерпретабельності (INP), зрозумілості (CMH), верифікованості (VFB), інтерактивності (INR);
- законність (LFL) – здатність ШІ відповідати законодавчим і нормативно-правовим актам;
- довірчоздатність (TST) – здатність ШІ, що характеризується ступенем впевненості користувача або іншої зацікавленої особи в тому, що ШІ відповідає вимогам і виконує функції у передбачуваний спосіб; її підхарактеристиками є: диверсність (DVS), функційна безпечність (SFT), точність (ACR), резильєнтність (RSL), робастність (RBS), захищеність (SCR).

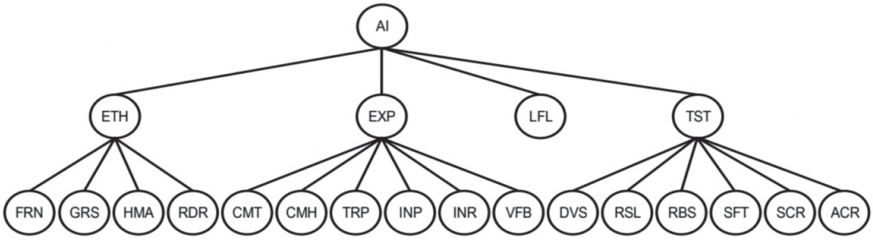



Рисунок 3.3 – Графова форма базової моделі якості системи [9]

Особливу увагу слід приділити довірчоздатності та пояснюваності, оскільки зростання складності систем (зокрема, для критичних застосувань) потребує гарантування виконання вимог, а саме здатності зрозуміти причини прийняття рішення ШІ. Це обумовлює важливість розроблення та впровадження методів, засобів створення, верифікації та розгортання eXplainable Artificial Intelligence (XAI) як сервісу (XAIaaS). Перелік і взаємовідношення характеристик XAIaaS наведено на рис. 3.4.

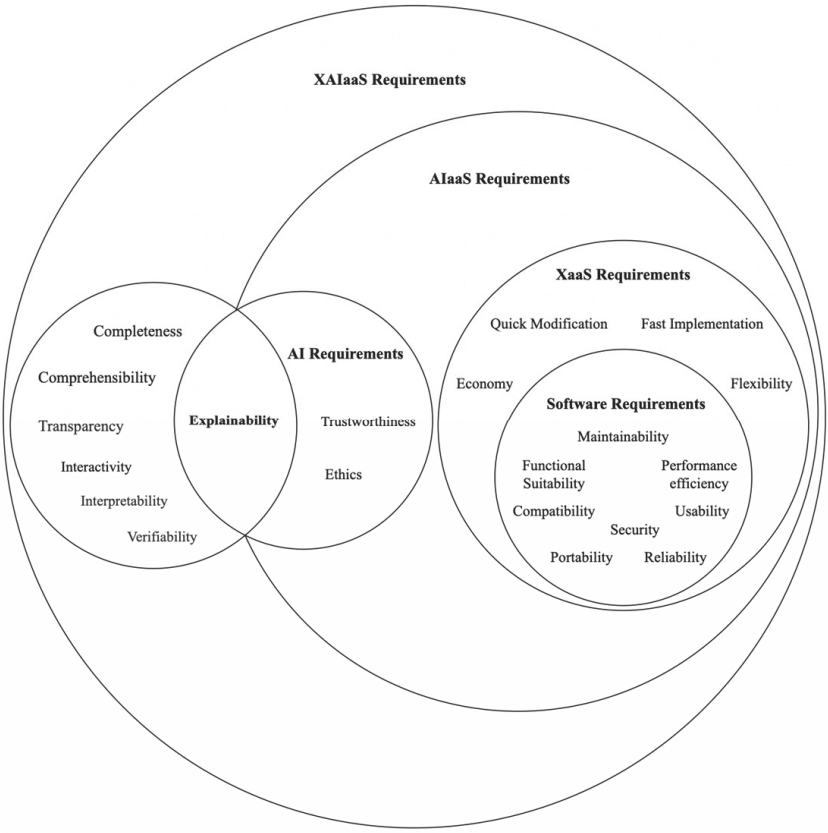



Рисунок 3.4 – Вимоги та характеристики XAIaaS

### 3.3.3. Особливості забезпечення пояснюваності штучного інтелекту

Особливості забезпечення пояснюваності ШІ. Для додавання пояснюваності моделі необхідно визначити, до якого типу моделей вона належить: скляної скриньки (Glass Box) чи чорної скриньки (Black Box). У

моделі Glass Box відомі всі параметри, є повне розуміння формування даних для висновку, які фактори та у якій мірі впливають на результат.

В моделі Black Box неможливо отримати доступ до так званого «вмісту» скриньки, внаслідок чого немає розуміння того, як працює модель та які показники є більш чи менш вагомими при визначенні результату. Водночас відомо, що модель Black Box має суттєві переваги порівняно з Glass Box, а саме більшу точність та ефективність, які підсилюються зі зростанням об'єму даних.

Залежно від типу моделі існує два підходи до реалізації XAI. Перший – розроблення сервісів з використанням інтерпретованих алгоритмів, зокрема Linear Regression, Logistic Regression, Decision Trees, Generalized Additive Models (GAMs), для розв'язання задач класифікації. Другий – надання пояснення алгоритмам, «внутрішня» робота яких є невизначеною. Відкриття скриньки надасть змогу об'єктивно висвітлити необхідну інформацію і зрозуміти причини прийняття кінцевого рішення системою. Алгоритмами для таких моделей є Tree Ensembles, Deep Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks.

### 3.4. Приклад розроблення моделі та метричного оцінювання якості XAIaaS

На підставі запропонованої у [9] базової моделі розглянемо приклад побудови моделі якості для конкретного AIaaS такого як: система виявлення зброї.

Приклад розроблення моделі та метричного оцінювання якості XAIaaS. З використанням запропонованих у роботі [9] базових і узагальнених моделей якості ШІ, платформ і систем ШІ розглянемо приклад побудови моделі якості для системи ШІ, призначеної для виявлення зброї. На рис. 3.5 наведено модель системи виявлення зброї (СВЗ) у людей на фото (відео), яку можна інтегрувати в присутній системи безпеки для припинення масової стрілянини та насильства. Вона є складовою базової моделі, де відповідні профільовані характеристики СВЗ марковані сірим кольором. Враховуючи особливості СВЗ, до множини суттєвих включено всі характеристики ШІ першого рівня (етичність ETH, пояснюваність EXP, довірчоздатність TST, законність LFL).

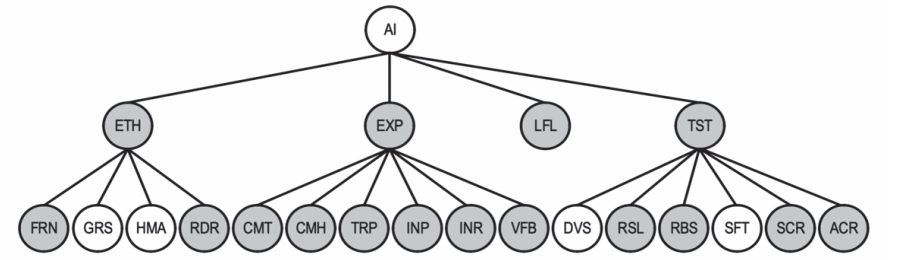



Рисунок 3.5 – Графова форма базової моделі якості СВЗ

До етичності включено дві з чотирьох підхарактеристик:

- справедливість (FRN) – оскільки важливо обробляти дані без упередженого ставлення до різних категорій людей (за статтю, расою, віком та ін.) [10];
- відшкодовуваність (RDR) – за умови, якщо все ж таки ШІ зробить помилку та з упередженістю поставиться до когось, повинна бути можливість відшкодування наслідків.

До пояснюваності включені всі підхарактеристики:

- транспарентність (TRP), зрозумілість (CMH) та інтерпретабельність (INP), необхідні для аргументації прийнятих важливих для людини рішень, тобто який саме об'єкт був визначений як зброя, тип об'єкта та його місце розташування;
- верифікованість (VFB) та завершеність (CMT) є невіддільними складовими сервісу, бо відсутність даних характеристик повністю унеможливлює використання СВЗ;
- інтерактивність (INR) є важливою з точки зору взаємодії з операційним персоналом.

Законність як характеристика має бути додана, оскільки система повинна відповідати законодавчим і нормативно-правовим актам.

Довірчоздатність представлена чотирма підхарактеристиками з шести:

- робастність (RBS) – її врахування обумовлюється потребою визначати наявність озброєння у неочікуваних місцях в різних умовах;
- захищеність (SCR) – з огляду на необхідність захищеності від атак, внаслідок яких можливі зміни, наприклад, вагових коефіцієнтів нейромоделі, що ідентифікує зброю;
- точність (ACR) – підхарактеристика критично важлива для СВЗ, тому що помилки системи будуть коштувати їй довіри;
- резильєнтність (RSL) – виникнення неспецифікованих порушень і відмов повинно коректно оброблятися ШІ у СВЗ.

### 3.5. Експериментальне оцінювання якості XAIaaS

Існуючі AIaaS рішення можна вважати узагальненими Black Box моделями через відсутність доступу до вихідного коду, отже й зрозуміти внутрішню будову та роботу сервісів неможливо. Але беручи за основу результати роботи систем, можна виділити параметри, що нададуть можливість оцінити відповідність сервісів критичним вимогам AI.

Серед усіх можливих сервісів найбільш що потребують пояснення є сервіси роботи з відео та зображеннями. Компанії Microsoft, Amazon, Google тощо, створили платформи обробки зображень з шаром абстракції, щоб спростити зручність використання та стимулювати розгортання без явного навчання моделі. Характеристики AI як Етичність та Пояснюваність можна дослідити на прикладах роботи Google Vision AI & Amazon Rekognition.

Визначення параметрів, що допоможуть оцінити Пояснюваність складається з наступних етапів:

- визначення функціонала обох сервісів;
- вибір логічно схожих функцій;
- вибір метрик для оцінки характеристик;
- формування тестових даних;
- порівняння результатів (отриманих метрик) обробки даних.

Перелік поширених функцій, що надають провайдерами зображено в табл. 3.2.

Таблиця 3.2 – Функції аналізу зображень, що надається хмарними провайдерами.

| **Функції** | **Google Vision AI** | **Amazon Rekognition** |
|---|---|---|
| Розпізнавання міток | ✓ | ✓ |
| Виявлення орієнтирів | ✓ | ✗ |
| Виявлення облич | ✓ | ✓ |
| Виявлення логотипів | ✓ | ✗ |
| Розпізнавання тексту | ✓ | ✓ |
| Розміщення об'єктів | ✓ | ✓ |
| Пошук пов'язаного вебу змісту за зображенням | ✓ | ✗ |
| Виявлення відвертого змісту | ✓ | ✓ |
| Визначення відомих осіб | ✗ | ✓ |
| Порівняння облич | ✗ | ✓ |

Згідно з документацією сервісів можна виділити схожі за функціональністю сервіси такі як: розпізнавання міток, облич, тексту, визначення місцезнаходження об’єктів та аналіз зображень (та для деяких відео) на наявність відвертого контенту.

Для вибору використання сервісів необхідно враховувати їх особливості та обмеження. Amazon Rekognition дозволяє завантажувати зображення лиже у форматах JPEG та PNG розміру не більше - 5 MB. Google Cloud Vision AI підтримує JPEG/PNG8/PNG24/GIF/AnimatedGIF (тільки перший фрейм) /BMP/WEBP/RAW/ICO/PD/TIFF та розмір не повинен перевищувати 20MB.

В межах дослідження можливостей AI сервісів було проведено експеримент націлений на пошук параметрів що б відповідали критичним вимогам XAIaaS. Серед них було обрано перевірити Пояснюваність.

Враховуючи дані, що повертаються сервісами при обробці зображень, найбільш відповідною характеристикою для надання пояснюваності AI є визначення місцезнаходження об’єкту. Обидві сервіси надають інформацію про місцезнаходження об'єктів (міток) у вигляді рамки, де “за їх думкою” знаходиться об’єкт. Для визначення облич або тексту присутні більш розширені значення, наприклад координати тієї чи іншої частини обличчя.

Для реалізації першого експерименту було написано програму, що використовує пошук різних видів зброї на зображеннях та їх місце розташування використовуючи обидва сервіси Amazon Recognition & Vision AI. Для пошуку використовувалися позначки: 'Bow and arrow', 'Racket' ,'Gun Barrel', 'Revolver', 'Cannon', 'Dagger', 'Knife', 'Rifle', 'Shotgun', 'Tank', 'Axe', 'Handgun', 'Sword', 'Gun', 'Missile', 'Air gun', 'Machine Gun', 'Weapon', 'Weaponry', 'Bomb', частина з яких знайдена на ієрархічній класифікації міток від Google Vision, зображено на рис. 3.6 та з переліку міток від AWS Rekognition AI.

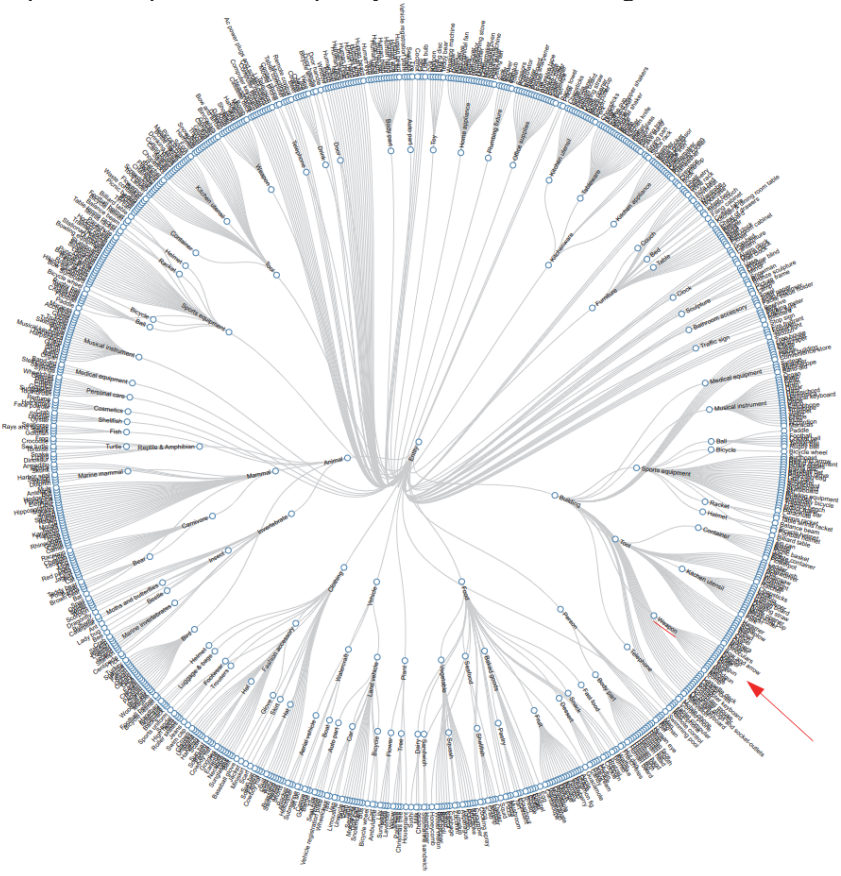


Рисунок 3.6 – Ієрархічна модель міток, доступних в Google Vision AI [11]

Тестовими даними були обрані 100 зображень з людьми зі зброєю. Результати аналізу зображень відображено в табл. 3.3.

Таблиця 3.3 – Результати аналізу зображень на наявність зброї сервісами Google Vision AI & Amazon Recognition

| | Кількість розпізнаних об'єктів зброї | Кількість розпізнаних об'єктів зброї з рамками | Упевненість у визначенні об'єктів | | |
|---|---|---|---|---|---|
| | | | Висока 90 - 100 % | Достатня 75 - 90 % | Низька 50 - 75 % |
| Amazon Rekognition | 217 | 52 | 141 | 38 | 38 |
| Google Vision AI | 66 | 4 | 5 | 46 | 15 |

У ході дослідження було виявлено, що із задачею розпізнавання зброї Amazon Rekognition порається краще ніж Google Vision AI, при чому як і просто з пошуком міток зброї, так і з її місце розташування на зображенні. Об'єктів зброї сервісом Amazon Rekognition було знайдено втричі більше аніж Google Vision AI, та впевненість у наявності зброї на зображеннях аналогічно у AWS сервісу значно вища ніж у Google, що можна побачити на рис. 3.7. Також було виявлено, що Google Vision AI має тенденцію плутати об'єкти зброї з іншими речами, підтвердження на рис. 3.8.

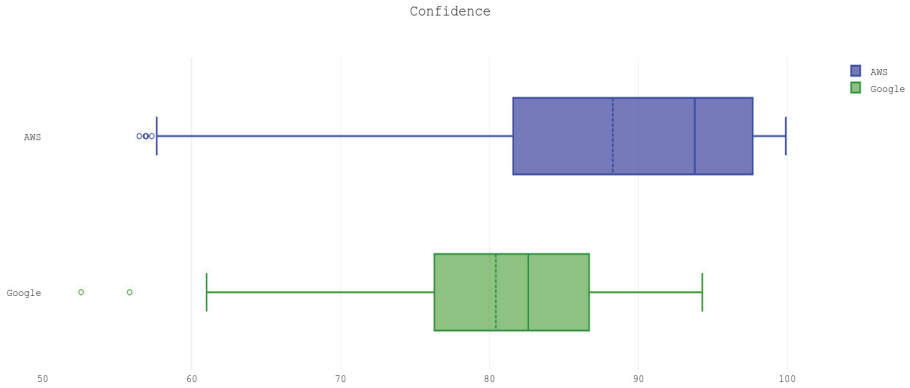



Рисунок 3.7 – Розподіл впевненості у прийнятих рішень щодо виявлення зброї

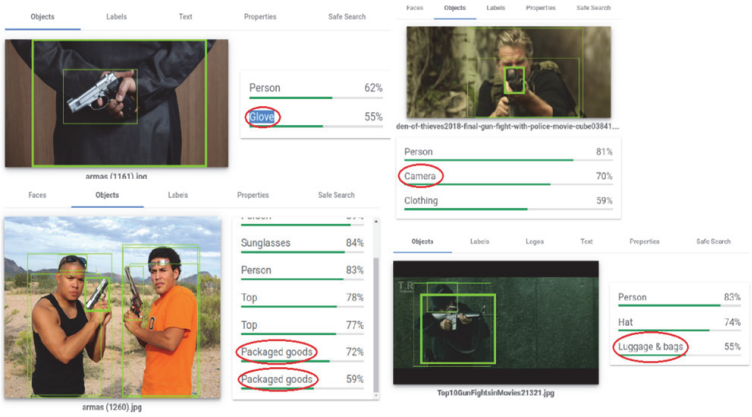



Рисунок 3.8 – Приклади неправильно визначених об’єктів замість зброї сервісом Google Vision AI

## 3.6. Висновки

Використання AI як сервісів є не тільки даниною часу - складовою потужного тренду Everything as a Service, але й об'єктивною необхідністю для багатьох застосувань, коли "на борту" системи неможливо або недоцільно розгортати потужні обчислювальні/аналітичні ресурси, що спонукає до подальших досліджень задач оптимізації розгортання AI ресурсів, сфери периферійних обчислень (edge computing).

Ефективне використання AIaaS, зокрема, в чутливих до рівня надійності і безпеки системах неможливе без перевіреного оцінювання ключових характеристик якості AI, а саме, пояснюваності, довірчоздатності, резильєнтності тощо. Запропоновані кейси оцінки якості AI продуктів на підставі моделей[9] надають змогу уточнювати вимоги до власне продукту і до системи - хмарної інфраструктури в цілому. Таким чином важливо впроваджувати інтегровану модель якості всієї системи надання AI послуг, забезпечуючи її рівноміцність.

Перевірка характеристик AI сервісів має доповнюватися експериментальними оцінками, краще за все, на реальних ресурсах. В статті проведено експериментальне оцінювання якості сервісів, наданих хмарними провайдерами на прикладах розв'язання побутових задач з урахуванням відповідати заданим не функційними характеристиками. Хмарні послуги ШІ мають попит, що підтверджує актуальність вдосконалення існуючих сервісів та впровадження нових, більш розширених нових. Як було з’ясовано в ході експерименту, готові сервіси гарно пораються із задачами загального призначення, але на цю мить створення універсального сервісу для вирішення усіх вузькоспеціалізованих потреб є все ще недосяжною метою. Важливий

висновок полягає у необхідності чіткого визначення меж аналітичного, експертного і експериментального оцінювання та розроблення варіантів поєднання цих методів для забезпечення достовірності оцінок та гарантування відповідності AI сервісів вимогам.

Література


1. Peter Elger, Eóin Shanaghy, AI as a Service, Shelter Island, NY, USA, Manning Publications Co, 2020, ISBN 9781617296154
2. 21 Artificial Intelligence Examples and Use Cases - Lasse Rouhiainen. Lasse Rouhiainen - International Keynote Speaker on Metaverse, Artificial Intelligence and Web3. URL: https://lasserouhiainen.com/21-ai-examples (дата звернення: 27.04.2023).
3. How APAC enterprises can scale up their AI initiatives. URL: https://media.bitpipe.com/io_10x/io_102267/item_1306461/E-guide_AI_in_APAC.pdf (дата звернення: 27.04.2023).
4. Pykes K. 5 AI Failures You Probably Should Know About. Medium. URL: https://towardsdatascience.com/5-ai-failures-you-probably-should-know-about-417ddebbc323 (дата звернення: 27.04.2023).
5. Kharchenko V., Fesenko H., Illiashenko O. Quality Models for Artificial Intelligence Systems: Characteristic-Based Approach, Development and Application. Sensors. 2022. Т. 22, № 13. С. 4865. URL: https://doi.org/10.3390/s22134865 (дата звернення: 27.04.2023).
6. Sebastian Lins, Konstantin D. Pandl, Heiner Teigeler, Scott Thiebes, Calvin Bayer, Ali Sunyaev, Artificial Intelligence as a Service, Springer International Publishing, 2021. Available: https://link.springer.com/content/pdf/10.1007/s12599-021-00708-w.pdf. Accessed on: July 24, 2022
7. ISO/IEC 25010. Available: https://iso25000.com/index.php/en/iso-25000-standards/iso-25010. Accessed on: July 24, 2022
8. Overview of Everything as a Service (XaaS). Available: https://www.geeksforgeeks.org/overview-of-everything-as-a-service-xaas/. Accessed on: July 24, 2022
9. Vyacheslav Kharchenko, Herman Fesenko, Oleg Illiashenko, Quality Models for Artificial Intelligence Systems: Characteristic-Based Approach, Development and Application, MDPI 2022. Available: https://doi.org/10.3390/s22134865. Accessed on: July 24, 2022
10. Shona Ghosh, Google AI will no longer use gender labels like 'woman' or 'man' on images of people to avoid bias. Available: https://www.businessinsider.com/google-cloud-vision-api-wont-tag-images-by-gender-2020-2. Accessed on: July 24, 2022
11. Google Vision API Labels. Available: https://storage.googleapis.com/openimages/2018_04/bbox_labels_600_hierarchy_visualizer/circle.html. Accessed on: July 24, 2022

# 4. МЕТОДИ ОЦІНЮВАННЯ ЯКОСТІ СИСТЕМ ДОПОВНЕНОЇ РЕАЛЬНОСТІ

**Є. О. Канарський, О. О. Орєхов, А. О. Стадник**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 4.1. Вступ

Доповнена реальність (англ. Augmented Reality, AR) – це технологія, що дозволяє інтегрувати віртуальні об'єкти у оточуючу нас реальність. Іноді як синоніми використовуються назви розширена реальність, поліпшена реальність, збагачена реальність. Вперше цей термін був запропонований у 1992 році інженером Томом Коделлом. Дослідник Рональд Азума в 1997 році визначав доповнену реальність як систему, що [1]:

- поєднує віртуальне і реальне;
- взаємодіє в реальному часі;
- працює в 3D.

Раніше за нього у 1994 Пол Мілграм та Фуміо Кішино описували доповнену реальність як частину змішаної реальності(MR) [2]. Точніше, вони позначають доповнену реальність як частину реально-віртуального континууму, який з'єднує повністю реальне середовище з віртуальним (рис. 4.1).


Рисунок 4.1 – Реально-віртуальний континуум

Хоч технологія доповненої реальності отримала широке поширення та почала активно розвиватись у різних сферах відносно недавно, сама технологія не є новою. Перше практичне використання доповнена реальність знайшла ще в 80-х роках минулого століття. Разом з початком використання AR-технологій постало питання про методи для оцінки якості таких систем. В наш час, коли доповнена реальність отримала значний розвиток у зв'язку з поширенням мобільних пристроїв, це питання стало ще більш актуальним.

Головною відмінністю систем доповненої реальності від «традиційних» програмних продуктів являється інтерфейс. На відміну від таких звичних елементів керування, як кнопки та повзунки тощо, AR-системи використовують інші способи забезпечення можливості взаємодії користувача з віртуальними елементами. Саме на цю особливість робиться акцент при дослідженні проблематики оцінки якості доповненої реальності та інших складових змішаної реальності.

### 4.2. Методи оцінки AR-систем на основі якості використання

Більша частина досліджень оцінки якості доповненої реальності присвячена якості використання (англ. usability) AR-систем, особливо інтерфейсу користувача. Перші дослідження стосувались оцінки якості використання віртуальних середовищ в цілому і представляли собою просте опитування користувачів віртуальної або доповненої реальності. При цьому відзначалися поширені проблеми, пов'язані зі зручністю використання та уявленням користувачів про роботу системи [3]:

- оманливі ознаки дій;
- очікування дії, яка неможлива;
- прихований функціонал;
- відсутній або неоднозначний зворотній зв'язок.

У дослідженні ці проблеми пов'язують з розташуванням елементів інтерфейсу та сприйняттям користувачами елементів змішаної реальності для орієнтуванням у віртуальному просторі. Хоч дана робота не пропонує власних методів оцінки якості систем доповненої реальності, вона позначила їх основні проблеми та стала основою для подальших досліджень.

#### 4.2.1. Покрокові методи

Одним з таких результатів став представлений у роботі [4] покроковий метод оцінювання (WEM). Даний метод являється вдосконаленою покроковою перевіркою на відповідність до поставлених вимог і включає наступні основні кроки:

- підготовка сценаріїв та контрольних списків для користувачів;
- проходження кожного сценарію з використання відповідних моделей;
- критика дизайну системи;
- реєстрація виявлених проблем;
- об'єднання проблем, виявлених в кожному сценарії, в загальних список;
- розстановка пріоритетів вирішення проблем на основі суб'єктивної оцінки.

Такий метод доволі простий у реалізації та дозволяє залучати до оцінки якості використання персонал без відповідної кваліфікації. Для підвищення точності отримуваних результатів пропонується повторний аналіз з більшою кількістю оцінювачів. За результатами перевірки, WEM може правильно визначити до 80,4% спостережуваних проблем. Упущені проблеми переважно пов'язані з труднощами навігації і не являються критичними. Загалом таких методів для оцінки якості використання віртуального середовища було створено досить багато. Їх класифікація стала предметом окремого дослідження (рис. 4.2) [5].

| Контекст оцінювання | | Залучення користувачів: Потребує | Залучення користувачів: Не потребує | Тип результатів | |
|---|---|---|---|---|---|
| | Загальні | • Формальне підсумкове оцінювання<br>• Post-hoc опитування | • Загальні моделі продуктивності для віртуальних середовищ | Кількісний | Тип результатів |
| | Загальні | • Неформальне підсумкове оцінювання<br>• Post-hoc опитування | • Евристичний аналіз | Якісний | Тип результатів |
| | Для конкретних застосунків | • Формальне оцінювання<br>• Формальне підсумкове оцінювання<br>• Post-hoc опитування | • Моделі продуктивності для конкретних програм віртуальних середовищ | Кількісний | Тип результатів |
| | Для конкретних застосунків | • Формальне оцінювання (формальне і неформальне)<br>• Post-hoc Опитування<br>• Інтерв'ю | • Евристичний аналіз<br>• Когнітивний покроковий метод | Якісний | Тип результатів |

Рисунок 4.2 – Класифікація методів дослідження якості використання віртуальних середовищ

Проте, не зважаючи на великий перелік доступних методів та їх ефективність при проведенні аналізу, залишалась актуальною потреба в більш економічно-ефективних методах інженерії якості використання [6] і адаптація їх до майбутніх вимог галузі [7].

4.2.2. Метод аналізу домена

Відокремлювати змішану реальність від віртуального середовища як самостійне явище для аналізу якості використання стали лише на початку XXI століття [8]. Одним з перших адаптованих до оцінки доповненої реальності методів став метод аналізу домену [9]. Запропонований процес аналізу складається з чотирьох основних дій:

 розробка варіантів використання – варіанти використання описують випадки використання, для яких призначена система;

 профілі користувачів – дозволяє інженерам зосередити зусилля на проектуванні для певної цільової групи;

 аналіз потреб користувачів – уточнює потреби користувачів та дає аналіз необхідних можливостей;

 аналіз завдань користувача – використовує набір методів для декомпозиції завдань користувачів та розуміння набору процедур, які користувач використає для досягнення мети. Цей аналіз корисний для встановлення показників ефективності, призначення цільових значень для метрик, визначення потенційних помилок та перевірки відповідності інтерфейсу до потреб користувача.

Дане дослідження стало першим випадком використання орієнтованого на користувача процесу проектування до AR-систем. Також існують методи з використанням підтримки комп'ютера [10]:

 I-DOVE (Interactive tool for Development Of Virtual Environment) – заснований на кількох наборах рекомендацій для розробки віртуальних середовищ;

 MAUVE – багатокритеріальна usability-матриця для отримання рекомендацій щодо навігації, взаємодії з об'єктами, дизайну тощо;

 гіпертекстовий прототип – інструмент для підтримки при проектуванні інтерфейсів віртуальних середовищ. Містить 45 загальних властивостей дизайну, які необхідні для нормального інтерфейсу.

### 4.2.3. Метод анкетування

Окрім комп'ютеризованих засобів також існують інші методи оцінки, основані на залученні користувачів до процесу розробки [11]:

 польове спостереження – складається зі спостереження за користувачем та збором інформації про його поведінку та результати діяльності при виконанні поставлених завдань. Дослідження проводиться в робочому середовищі користувача. Дослідник має спостерігати за робочим процесом та роби-ти записи, на основі яких ставиться оцінка;

 інтерв'ю – метод виявлення потреб та індивідуальних проблем потенційних користувачів за допомогою інтерв'ю;

 анкетування – застосовується командою дизайнерів для з'ясування, як розроблювана система буде використовуватись певною групою користувачів;

 аналіз контексту використання – описує відповідні характеристики користувача (ISO 9421-11). Включає описи користувача, виконувані завдання, використовуване обладнання та робоче середовище. Ця інформація має важливе значення для розробки інших методів оцінки якості використання;

 збори вимог – виконується після аналізу контексту використання для аналізу результатів;

 експертна оцінка за критеріями – оцінювання експерта на основі заздалегідь встановлених критеріїв;

 оцінка зручності використання – у тесту-ванні бере участь група користувачів, яка має виконувати невеликі завдання на основі підготовлених коротких сценаріїв. Після ви-конання завдань проводиться інтерв'ю і заповнюється анкета;

 діаграма спорідненості – простий і економічно ефективний метод сортування великих обсягів даних. Використовується для групування різних вимог користувачів.

Дані методики існують вже давно і не забезпечують високу точність результатів, але все ще залишаються актуальними через свою простоту використання.

Повноцінне дослідження якості систем доповненої реальності почалось з 2007 року, коли мобільні пристрої стали досить поширеними. Саме мобільні пристрої стали платформою для розповсюдження AR-систем за межами спеціально обладнаних робочих місць. У роботі [12] пропонується методологія оцінювання якості використання для прототипу AR-бінокуляра PRISMA. Предметом оцінки названо поведінку користувача з новими туристичними технологіями. Оцінювання проводиться за кількісними і якісними показниками. Основними цілями кількісного аналізу являється оцінювання поведінки та підтвердження доданої вартості учасниками тестування. Аналіз проводив-ся на основі анкети із 30 запитань, заповнених одразу після використання прототипу. Якісний аналіз збирає дані за допомогою методів безпосереднього спостереження, інтерв'ю та дослідження письмових документів. Його метою є визначення потреб і бажань користувачів подібних туристичних AR-систем. Такий підхід до оцінювані якості використання може вказати на основні недоліки системи та дати підказки для подальшого вдосконалення. В опитуваннях часто беруть участь зацікавлені представники відповідної сфери, що впливає на результати. Проте оцінювання проходить на основі відгуків фокус-групи з невеликою кількістю учасників, а залучати більше людей не дозволяє потік користувачів на тестовій локації. Такі результати не можна назвати об'єктивними або точними, але такий метод тестування залишається актуальним завдяки простоті та швидкості отримання результатів.

Більш комплексне дослідження вкладено в роботі [13]. Головною метою дослідження являється розробка методу інженерії якості використання, орієнтованого на користувачів та включення його у життєвий цикл розробки. Таким чином пропонується поліпшити дизайн інтерфейсу додатків доповненої реальності. Як і в роботі [12], тут використовуються дослідження поведінки користувачів, проте результати додатково підкріплені розрахунками кореляції, впливу стилів тексту та алгоритмів малювання на помилку. Також до уваги було взято проблеми з освітленням, налаштування та інші фактори, що

впливали на користувача. В результаті був використаний той самий метод опитування користувачів, але завдяки додатковим розрахункам результати можна вважати більш валідними. Проблемою дослідження являється те, що воно направлено на дослідження активних стилів малювання для доповненої реальності. Сучасні AR-системи не обмежуються текстовими даними, тому потрібні додаткові дослідження і розрахунки, без яких робота являється застарілою.

Схожі тези можна побачити в публікації [14] від 2009 року. Розглядаючи методи опитування, перевірки та тестування для оцінки якості використання AR-інтерфейсів, автор відзначає необхідність враховувати відмінність доповненої реальності від інших систем. Характеристика методів перевірки та опитування відповідають тим, що описані в дослідженнях [12] і [13] відповідно. Метод тестування описується як основний метод оцінки якості використання для систем доповненої реальності, але за описом він майже ідентичний до методу перевірки і не має прикладів використання.

### 4.2.4. Метричний метод

Дослідження якості використання не обмежується стандартизованими опитуваннями. Для організації, стурбованих покращенням взаємодії з користувачами, відстеження та вимірювання якості використання є постійною проблемою. Вирішенням проблеми стандартизації якості використання займається в тому числі IT-відділ компанії Intel [15]. Для стандартизації була обрана п'ятибальна шкала Лайкерта – так звана шкала зручності використання системи (SUS). Однак просто адаптувати SUS для роботи з доповненою реальністю виявилось неможливим, тому було прийнято рішення пов'язати пул потенційних елементів зі стандартом ISO 9241-11. Загалом було розроблено 12 потенційних пунктів, за якими повинна проводитись оцінювання. Всі пункти були поділені на три категорії – ефективність, результативність(дієвість) і задоволеність (табл. 4.1).

Таблиця 4.1– Використані потенційні елементи

| **Компоненти usability** | **Потенційні елементи** |
|---|---|
| Ефективність | [Цей елемент] зберігає мені час |
| | Я схильний роботи помилки [в цій системі] |
| | Я не роблю помилок [з цією системою] |
| | Я маю витратити багато часу на виправлення [з цією системою] |
| Дієвість | [Ця система] дозволяє мені виконувати мої завдання |
| | Для моїх завдань мені потрібна система з більшою кількість функцій |
| | Мені не потрібні доповнення [для цієї системи] |
| | Можливості [цієї системи] не відповідають моїм вимогам |

Продовження табл. 4.1

| | |
|---|---|
| Задоволеність | Я задоволений [цією системою] |
| | Я краще користувався б чимось іншим замість [цієї системи] |
| | Маючи вибір, я обрав би [цю систему] замість інших |
| | Використання [цієї системи] було розчаруванням |

Після декількох раундів тестувань, була проведена кореляція показників. Були внесені значні зміни для покращення балансу та усунення плутанини з елементами (табл. 4.2).

Таблиця 4.2 – Компоненти якості використання (usability)

| **Usability-компонент** | **Потенційні елементи** |
|---|---|
| Дієвість | Можливості [цієї системи] відповідає моїм вимогам |
| Задоволеність | Використання [цієї системи] виклика розчарування |
| Загальність | [Ця система] легка у використанні |
| Ефективність | Я витрачаю багато часу на виправлення помилок [в цій системі] |

Отримані метрики якості використання для користувацького досвіду було визнано як надійною, валідною та чутливою альтернативою традиційної шкали SUS. Самі автори досліджень відзначають компактний розмір отриманих метрик та потенційну можливість використовувати їх в інших етапах життєвого циклу. Разом з тим відзначається високий рівень кореляції між показниками. Проте компактність описаних в роботі [15] метрик не дає всебічну оцінку досліджуваної системи, а лише спирається на суб'єктивну оцінку користувачів та їх власний досвід. Результати дослідження можна використати для подальших розробок в напрямку оцінки якості використання.

Для оцінки систем доповненої реальності можна використовувати SUS без додаткових модифікацій. У дослідженні [16] оцінюється розроблена інтерактивна AR-система для навчання стосовно збереження риб на Тайвані. У частині оцінки системи в основному оцінювалась якості використання системи з точки зору кінцевого користувача. Саме опитування складається з 10 питань, які оцінюються за 5-бальною шкалою:

- ☐ я вважаю, що хочу використовувати цю систему частіше;
- ☐ я вважаю, що система надто складна;
- ☐ я вважаю, що система проста у використанні;
- ☐ я вважаю, що мені знадобиться технічна підтримка для використання цієї системи;
- ☐ я вважаю, що окремі функціє цієї системи добре інтегровані;
- ☐ я вважаю, що в цій системі багато суперечностей;

 я вважаю, що більшість людей швидко навчиться користуватись цією системою;

 я вважаю, що система дуже громіздка для використання;

 я вважаю, що відчуваю себе дуже впевнено при користуванні системою;

 я вважаю, що мені потрібно багато чому навчитися для роботи з цією системою.

Анкета заповнюється учасниками опитування після завершення роботи з досліджуваною системою, після чого розраховується середній бал та медіана. Для розглянутої роботи ці значення становлять 78 і 66 балів відповідно, що вказує на придатність системи до використання. Учасники опитування також відмітили зручність використання системи. Загалом можна сказати, що SUS відноситься до вже розглянутої методики опитування за допомогою анкетування. Головною відмінністю SUS являється чітко визначений, сталий перелік питань. З точки зору проведення оцінки, він являється ефективним, економічним за часом та простим у використанні. Головним його мінусом являється суб'єктивність отриманих результатів.

### 4.2.5. Евристичний метод

Окрім опитувань та перевірки за допомогою метрик, популярною методикою оцінки якості використання являється евристичний метод. Його суть полягає в тому, що група експертів перевіряє дизайн інтерфейсу за допомогою набору характеристик. Евристичне оцінювання просте у виконанні, дешеве та доволі ефективне. За його допомогою не можна виявити всі проблеми, але переважна більшість основних і більшість другорядних недоліків стануть явними. Процеси перевірки якості використання за допомогою евристичної оцінки добре задокументовані і мають багато публікацій, що описують використання цих методів. Разом із ростом популярності AR-систем постало питання оцінки якості використання, в тому числі за допомогою евристичного методу. Оскільки традиційні евристики не підходять для оцінки доповненої реальності, в роботі [17] пропонується методологія створення нових евристик якості використання для кожного конкретного випадку:

 дослідницький етап – збір бібліографічного матеріалу, конкретні програми та їх характеристики, загальні та/або пов'язані евристики;

 описовий етап – визначення найважливіших характеристик раніше зібраних даних та формалізація основних понять;

 кореляційний етап – визначення характеристик, які повинні мати евристики якості використання на основі традиційних евристик та аналізу існуючих прикладів;

 пояснювальний етап – формальне визначення набору запропонованих евристик за допомогою стандартного шаблону;

 етап валідації – перевірка нових евристик за допомогою експериментів, виконаних на обраних тематичних дослідженнях, доповнених користувацькими тестами;

 етап уточнення – уточнення результатів на основі відгуків, отриманих на етапі перевірки.

Представлений алгоритм універсальний і підходить для будь-якої системи. Для перевірки його придатності в розглянутій роботі та досліджуваних greed-систем було розроблено 12 нових евристик, згрупованих у три категорії: дизайн та есте-тика, навігація, помилки та довідка. Результати роботи були підтримані School of Informatics Engineering of the Pontifical Catholic University of Valparaiso, членами "UseCV" Research Group та IDIS Research Group of University of Cauca. Загалом отримана таким чином евристики мають загальні для даного методу оцінки якості використання недоліки – можна пропустити деякі малопомітні проблеми. Проте недоліки компенсуються дешевизною, швидкість та простотою методу.

Крім вищевказаних, існує ще одна причина популярності використання евристичних досліджень. Головною платформою для додатків доповненої реальності являються смартфони та планшети. Через велику кількість способів використання AR у додатках різного призначення. Таке різноманіття ускладнює стандартизацію, тому простіше використати евристичне дослідження. У дослідженні [18] розглядаються принципи якості використання для AR-додатків на смартфоні. Метою дослідження стала розробка принципів якості використання для розробки та оцінки мобільних додатків. Розробка проводилась на основі аналізу існуючих досліджень методів евристичної оцінки, принципів проектування систем доповненої реальності, вказівок щодо інтерфейсів портативних мобільних пристроїв і принципів зручності використання матеріального інтерфейсу користувача. В результаті було розроблено 22 принципи, що були розділені на п'ять різних груп (рис. 4.3).

На основі отриманих результатів було проведено дослідження декількох мобільних додатків на операційній системі Android. Оскільки тестування проводилось в лабораторних умовах, результати тестування із залученням користувачів можуть відрізнятись. При спробі розробити аналоги до вже існуючих додатків з використанням отриманих рекомендацій, показники якості використання вдалось покращити. Таким чином було підтверджено валідність результатів даного дослідження. Слід враховувати, що дане дослідження було обмежене трьома мобільними застосунками, використовуваними лише на території Кореї. Тестування проходило в лабораторних умовах на смартфонах Android, тож на iOS результати не перевірялись. При дослідження використовувались дисплеї стаціонарних смартфонів, тож масштабування інтерфейсу не біло враховане.

Також розробці евристик для систем доповненої реальності присвячена робота [19]. У дослідженні проводиться розробка набору універсальних евристик для оцінки якості використання AR не тільки на мобільних, а й на

спеціалізованих гаджетах і платформах. У своїй роботі автор про-водить дослідження методів евристичної оцінки та їх застосування при тестуванні додатків доповненої реальності, а також модифікує деякі вже відомі. В результаті ним було запропоновано список з шести евристик для оцінки AR-додатків, а саме:

- методи взаємодії та контроль;
- презентація віртуальних об'єктів;
- зв'язок між віртуальними об'єктами і реальним світом;
- інформація, пов'язана з віртуальними об'єктами;
- придатність і контексті використання;
- фізичний контроль використання.

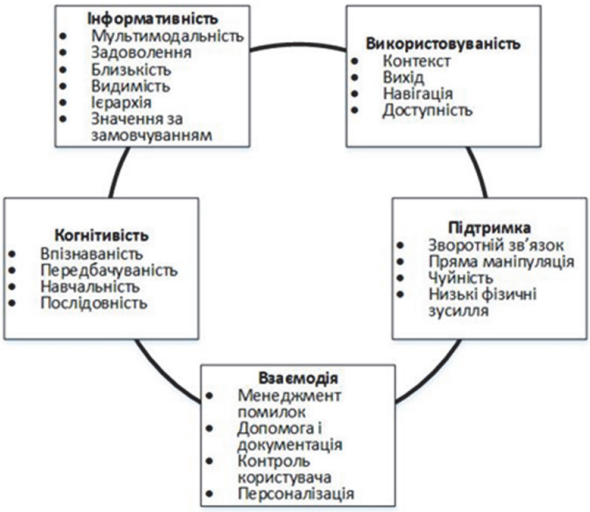



Рисунок 4.3 – Структуровані принципи якості використання

Даний список представляє собою універсальний набір на основі повторюваних і частих проблем, що зустрічаються при роботі з доповненою реальністю, тому їх можна назвати універсальними. Однак така універсальність не гарантує точність результатів оцінки. При розробці евристик для кожного конкретного випадку, їх перелік може становити кілька десятків. Це добре видно на прикладі двох попередніх розглянутих публікаціях. Тому автор даної роботи пропонує під час евристичного налізу також використовувати інші евристики, наприклад такі як запропоновані Якобом Нільсеном. Також автор не проводив перевірку отриманих результатів на практиці, тому при використанні даний набір евристик може потребувати корегувань в кожному окремому випадку.

В останніх публікаціях чітко можна побачити наступну тенденцію: не просто досліджуються та пропонуються методи оцінки якості використання AR-застосунків, а й враховуються сфера їх застосування. Наприклад, роботи [20] і [21] присвячені дослідженню принципів якості використання у додатках доповненої реальності для дітей дошкільного віку та організації освітнього процесу відповідно. Ці два дослідження проводились різними групами з різних університетів з різницею у два роки, але їх структура майже ідентична. Першим кроком досліджується наявна література. Крім загальних понять та методів оцінки якості використання, описуються та зводяться у таблицю характерні проблеми, з якими стикаються діти або студенти. Отримані результати використовуються для розробки принципів якості використання, з яких видаляються дублікати і комбінуються схожі. Після видалення і об'єднання дублікатів, визначені принципи діляться на групи та проводиться аналіз головних компонентів. Нарешті, в результаті аналізу були отримані 23 окремих принципи якості використання для розробки AR-додатків для дітей дошкільного віку у роботі [20] та навчання студентів у роботі [21]. На основі отриманих принципів якості використання автори досліджень планують провести розробку застосунків та використати їх для проведення евристичного аналізу.

### 4.3. Оцінка якості AR-систем на основі користувацького досвіду

Окрім якості використання, існують інші методи оцінки якості систем доповненої реальності. Оскільки такі сервіси пропонують новий вид взаємодії , для забезпечення успіху їм також потрібно розуміти очікування і потреби майбутніх користувачів. Оцінювання цих факторів та їх вплив на якість AR-систем здійснюється за допомогою дослідження досвіду користувача (англ. User Experience, UX).

Як і у випадку з дослідженням якості використання, у перших дослідженнях використовувався метод інтерв'ю для розуміння уявлень та вимог потенційних користувачів до AR-систем. У роботі [22] проводиться опитування та систематизація отриманих результатів для різних областей застосування. Опитуваними стали відвідувачі в торгових центрах, оскільки там завжди багато відвідувачів, розміщуються різноманітні магазини та проводяться розважальні заходи. На відміну від більшості подібних досліджень, дана робота не розглядає системи доповненої реальності в рамках конкретної предметної області. Натомість в роботі досліджуються ставлення користувачів до AR-систем, уявлення про їх роботу та загальне відношення до використання технологій та обміну інформацією. Автори пояснюють такий вибір напрямку досліджень тим, що вивчення UX на момент написання статті знаходяться ще в початковому стані, а також пошуком можливість виділити специфічні для AR характеристики UX. Наступним кроком досліджень автори

називають розробку конкретних показників для оцінки AR-систем на основі теорій і концепцій UX.

Більш розгорнуто результати даного дослідження та концепції UX в цілому представлено в дисертації [23] все того ж автора. Окрім методу інтерв'ю, використаного в роботі [22], в дисертації для збору інформації про потреби і очікування потенційних користувачів використовуються результати онлайн-опитувань. Завдяки отриманим відповідям автор отримав уявлення про те, який досвід користувачі очікують отримати від користування додатками доповненої реальності, та систематизувати очікуваний UX за наступними категоріями:

- інструментальний – базується на цінності технології для користувача в якості інструменту для полегшення діяльності;
- когнітивний – пов'язаний з думками та можливістю AR-сервісу задовольняти потреби користувачів у нових знаннях;
- емоціональний – пов'язаний з можливістю AR-сервісу викликати в користувача таку емоційну реакцію (задоволення, радість, захоплення, емоційне збудження, ностальгія);
- сенсорний – впливає на сприйняття користувачем оточуючого середовища та інтерактивність, залежить від здатності AR-системи впливати на естетичні почуття;
- соціальний – виникає при взаємодії користувача з людиною, безпосередньо пов'язаною з тою ж AR-системою;
- мотиваційний – стимулює користувача досягти якоїсь мети за допомогою технології.

Даний перелік описує досвід, до якого прагнуть досвідчені та потенційні користувачі AR-системи. Відповідно, розробники при проектуванні мають це враховувати і намагатись зробити такий досвід можливим. Але UX, не охоплює всього, що слід враховувати при проектуванні і не може прогнозувати реальний досвід, отримуваний користувачами в процесі роботи. У висновках до дисертації автор зазначає, що використання UX в конкретних областях являється складною задачею. Люди час-то мають абстрактні або узагальнені потреби та очікування від роботи з доповненою реальністю, які можна віднести і до інших складових змішаної реальності. Така відсутність конкретики теоретично може підштовхнути до більш розширених досліджень UX. В подальшому дана дисертація стала основою для дослідження автором концепції та суб'єктивних показників оцінки UX мобільних систем доповненої реальності.

В останніх дослідженнях UX розглядається більш комплексно. Якщо в попередніх дослідженнях розглядався виключно сприйняття і реакція користувачів на використання систем доповненої реальності, то в сучасних дослідженнях також беруться до уваги сторонні фактори, що можуть вплинути на користувацький досвід. В роботі [24] пропонується цілісна концептуальна

модель UX з 10 компонентів (рис. 4.4), вилучених із вже існуючих моделей – UXIVE Model.

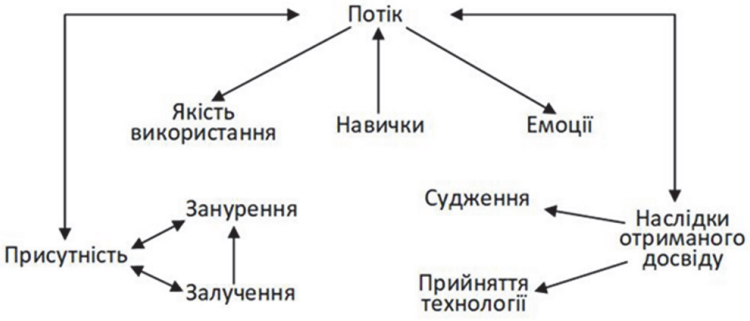



Рисунок 4.4 – Модель UXIVE

Модель UXIVE описує компоненти, з яких формується UX, та відношення між ними. Результати підкріплюються результатами опитування більш ніж 150 користувачів AR-систем. Автори справедливо зазначають, що отримані результати не являються абсолютно точними, оскільки в опитуванні брали участь обмежена кількість користувачів однієї AR-системи, половина з яких належить до однієї вікової групи. Але навіть з переліченими недоліками, дане дослідження являється першим, де вивчаються взаємозв'язки компонентів UX між собою, їх вплив на користувача і розвиток технологій доповненої реальності.

### 4.4. Оцінка якості AR-систем на основі візуальної складової

Крім розглянутих вище, існує ще велика кількість досліджень, пов'язаних з різними аспектами якості доповненої реальності. Ці публікації поєднують кілька напрямків досліджень або розглядають проблему під незвичним кутом, тому часто їх важко систематизувати.

До таких методів належить в тому числі оцінка якості зображення. Дослідження в цьому напрямі почались зовсім нещодавно, тому публікацій дуже мало. В них розглядають візуальну складову систем доповненої реальності як один з головних факторів, що впливає на користувацький досвід та якість системи в цілому. Так, наприклад, в дослідженні [25] пропонується метрика для оцінки якості сприйняття зображень на основі теорії візуальної плутанини. Автори дослідження розробили базу даних зображень доповненої реальності, а також відповідні суб'єктивні оцінки якості. В результаті своєї роботи автори отримали декілька метрик для оцінки впливу візуальної плутанини, а також запропонували метрику для оцінки якості сприйняття

зображень. Хоч дане дослідження не спрямоване на оцінку якості AR-систем в цілому, але зображення являється важливою складовою частиною доповненої реальності та має значний вплив на користувацький досвід. Тому дане дослідження та його результати являються важливими для подальших досліджень, пов'язаних з оцінкою якості доповненої реальності та AR-комунікацією.

Але яким би якісним не було створюване системою доповненої реальності зображення, воно сильно залежить від дисплею, на якому буде відтворюватись. В роботі [26] автори пропонують обчислювальну модель для оцінки якості зображень дисплеїв доповненої реальності. Дана модель дозволяє швидко оцінити якість зображення та визначити компроміси між конфігураціями дисплеїв, такими як яскравість, однорідність, роздільна здатність та рівень шуму. Ця робота підтримує розробку та оптимізацію дисплеїв доповненої реальності без обширних вимірювань або ресурсоємних досліджень сприйняття.

## 4.5. Висновок

В даній роботі проводиться огляд існуючих публікацій на тему оцінки якості доповненої реальності. Публікації можна систематизувати за напрямками досліджень. Найбільша кількість досліджень присвячена якості використання, оскільки за допомогою цього показника найлегше оцінити привабливість для користувача розроблюваного програмного продукту. Частіше всього для проведення оцінювання використовується метод евристичного дослідження. Такий вибір методики оцінки зумовлений тим, що якість використання являється суб'єктивною характеристикою і використання такого ж суб'єктивного методу оцінки здається логічним. На даний момент можна помітити що дослідження якості використання проходять в двох напрямках – перший зосереджений на визначенні універсальних евристик для будь-яких AR-додатків, другий передбачає розробку окремого набору евристик для кожної предметної області або окремого випадку. При цьому у другому випадку для покращення результатів оцінки часто додатково застосовуються універсальні евристики.

Також якість використання являється важливою складовою інших досліджень якості AR, таких як користувацький досвід або моделі на основі стандарту якості програмного забезпечення ISO-25010. Можна сказати, що якість використання являється ключовим параметром оцінки якості доповненої реальності, який доповнює інші параметри якості. Таким чином цей показник стане об'єднуючим при об'єднанні актуальних досліджень UX та вже існуючих метрик якості програмного забезпечення в єдину модель якості систем доповненої реальності.

На даний момент дослідження та створення моделей якості для систем доповненої реальності знаходиться на початковому етапі, тому публікацій не

цю тему майже немає. Це в значній мірі пов'язано з тим, що AR лише відносно нещодавно почали масово поширювати на смартфонах. AR-технології все частіше зустрічаються в нашому повсякденному житті, тому тема розробки моделей і метрик оцінки якості доповненої реальності являється актуальною і важливою.

Література

1. Azuma R. T. A Survey of Augmented Reality [Electronic resource] / Ronald T. Azuma // Presence: Teleoperators and Virtual Environments. – 1997. – Vol. 6, no. 4. – P. 355-385. – DOI: https://doi.org/10.1162/pres.1997.6.4.355

2. Milgram P. A Taxonomy Of Mixed Reality Visual Displays [Electronic resource] / Paul Milgram, Fumio Kishino // IEICE Transactions on Information Systems. – 1994. – E77D, no. 12. – P. 1321–1329. – Mode of access: https://cs.gmu.edu/~zduric/cs499/Readings/r76JBo-Milgram_IEICE_1994.pdf

3. Kaur K. Designing virtual environments for usability [Electronic resource] : Electronic Thesis or Dissertation / Kaur Kulwinder. – [S. l.], 1998. – Mode of access: http://openaccess.city.ac.uk/7567/

4. Sutcliffe A. G. Evaluating the usability of virtual reality user interfaces [Electronic resource] / A. G. Sutcliffe, K. Deol Kaur // Behaviour & Information Technology. – 2000. – Vol. 19, no. 6. – P. 415-426. – DOI: http://dx.doi.org/10.1080/014492900750052679

5. Bowman D. A. A Survey of Usability Evaluation in Virtual Environments: Classification and Comparison of Methods [Electronic resource] / Doug A. Bowman, Joseph L. Gabbard, Deborah Hix // Presence: Teleoperators and Virtual Environments. – 2002. – Vol. 11, no. 4. – P. 404-424. – DOI: https://doi.org/10.1162/105474602760204309

6. Gabbart J. L. Usability Engineering of Virtual Environments[Electronic resource] /Joseph L. Gabbard, Deborah Hix // Handbook of Virtual Environments (1st Edition). – Boca Raton, CRC Press, 2002. – P. 681-699. DOI: https://doi.org/10.1201/9780585399102

7. Tromp J. G. Systematic Usability Evaluation and Design Issues for Collaborative Virtual Environments [Electronic re-source] / Jolanda G. Tromp, Anthony Steed, John R. Wilson // Presence: Teleoperators and Virtual Environments. – 2003. – Vol. 12, no. 3. – P. 241–267. – DOI: https://doi.org/10.1162/105474603765879512

8. Gabbard J. L. Researching Usability Design and Evaluation Guidelines for Augmented Reality (AR) Systems [Electronic resource] / Joseph L. Gabbard. – Mode of access: https://www.rkriz.net/sv/classes/ESM4714/Student_Proj/class00/gabbard/index.html

9. Usability engineering: domain analysis activities for augmented-reality systems [Electronic resource] / Joseph Gabbard [et al.] // Electronic Imaging 2002,

San Jose, CA / ed. by A. J. Woods [et al.]. – [S. l.], 2002. – DOI: http://dx.doi.org/10.1117/12.468073

10. Bach C. Obstacles and Perspectives for Evaluating Mixed Reality Usability [Electronic resource] / Cédric Bach, Dominique L. Scapin // ResearchGate. – Mode of access: https://www.researchgate.net/publication/221104007_Obstacles_and_Perspectives_for_Evaluating_Mixed_Reality_Usability

11. Träskbäck M. User requirements and usability of mixed reality applications [Electronic resource] : Licentiate thesis / Träskbäck Marjaana. – Helsinki, 2004. – 109 p. – Mode of access: https://aaltodoc.aalto.fi/bitstream/handle/123456789/48/urn012717.pdf?sequence=1&isAllowed=y

12. Alzua-Sorzabal A. An Experimental Usability Study for Augmented Reality Technologies in the Tourist Sector [Electron-ic resource] / Aurkene Alzua-Sorzabal, María Teresa Linaza, Marina Abad // Information and Communication Technologies in Tourism 2007. – Vienna. – P. 231-242. – DOI: http://dx.doi.org/10.1007/978-3-211-69566-1_22

13. Gabbard J. L. Usability Engineering for Augmented Reality: Employing User-Based Studies to Inform Design [Electronic resource] / J. L. Gabbard, J. E. Swan // IEEE Transactions on Visualization and Computer Graphics. – 2008. – Vol. 14, no. 3. – P. 513-525. – DOI: https://doi.org/10.1109/TVCG.2008.24

14. Kostaras N. N. Assessing the Usability of Augmented Reality Systems / Nektarios N. Kostaras, Michalis N. Xenos // 13th Panhellenic Conference on Informatics, Corfu, 10–12 September 2009. – Los Alamitos, 2009. – P. 197–201

15. Finstad K. The Usability Metric for User Experience [Electronic resource] / Kraig Finstad // Interacting with Computers. – 2010. – Vol. 22, no. 5. – P. 323-327. – DOI: http://dx.doi.org/10.1016/j.intcom.2010.04.004

16. Lin H.-C. K. Establishment And Usability Evaluation Of An Interactive Ar Learning System On Conservation Of Fish [Electronic resource] / Hao-Chiang Koong Lin [et al.] // The Turkish Online Journal of Educational Technology. – 2011. – Vol. 10, no. 4. – P. 181–187. – Mode of access: https://files.eric.ed.gov/fulltext/EJ946626.pdf

17. Rusu C. A Methodology to Establish Usability Heuristics [Electronic resource] / Cristian Rusu [et al.] // The Fourth Inter-national Conference on Advances in Computer-Human Interactions, Gosier, 23–28 February 2011. – [S. l.], 2011. – P.59–62. – Mode of access: https://www.researchgate.net/publication/229040164_A_Methodology_to_establish_usability_heuristics

18. Ko S. M. Usability Principles for Augmented Reality Applications in a Smartphone Environment [Electronic resource] / Sang Min Ko, Won Suk Chang, Yong Gu Ji // International Journal of Human-Computer Interaction. – 2013. – Vol. 29, no. 8. – P. 501-515. – DOI: https://doi.org/10.1080/10447318.2012.722466

19. Kalalahti J. Developing Usability Evaluation Heuristics For Augmented Reality Applications [Electronic resource] : Masters thesis / Kalalahti Joanna. – Lappeenranta, 2015. – 90 p. – Mode of access: https://lutpub.lut.fi/bitstream/handle/10024/103081/masters_thesis_joanna_kalalahti_31122014.pdf?sequence=2

20. Tuli N. Usability Principles for Augmented Reality based Kindergarten Applications [Electronic resource] / Neha Tuli, Archana Mantri // Procedia Computer Science. – 2020. – Vol. 172. – P. 679-687. – DOI: https://doi.org/10.1016/j.procs.2020.05.089

21. Al-Obaidi A. Usability Principles for Augmented Reality Applications in Education [Electronic resource] / Arwa Al-Obaidi, Master Prince // International Journal of Computer Science and Network Security. – 2022. – Vol. 22, no. 1. – P. 49–54. – DOI: http://dx.doi.org/10.22937/IJCSNS.2022.22.1.8

22. Expected user experience of mobile augmented reality services: a user study in the context of shopping centres [Electronic resource] / Thomas Olsson [et al.] // Personal and Ubiquitous Computing. – 2011. – Vol. 17, no. 2. – P. 287-304. – DOI: http://dx.doi.org/10.1007/s00779-011-0494-x

23. Olsson T. User Expectations and Experiences of Mobile Augmented Reality Services [Electronic resource] / Thomas Olsson // Tampere University of Technology. – 2012. – P. 1–102. – Mode of access: https://urn.fi/URN:ISBN:978-952-15-2953-5

24. Towards a Model of User Experience in Immersive Virtual Environments [Electronic resource] / Katy Tcha-Tokey [et al.] // Advances in Human-Computer Interaction. – 2018. – Vol. 2018. – P. 1–10. – DOI: https://doi.org/10.1155/2018/7827286

25. Duan H. Augmented Reality Image Quality Assessment Based on Visual Confusion Theory [Electronic resource] / Huiyu Duan, Lantu Guo, Wei Sun, Xiongkuo Min, Li Chen, Guangtao Zhai // 2022 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Bilbao, Spain, 15-17 June 2022. –2022. – DOI: https://doi.org/10.1109/BMSB55706.2022.9828671

26. Zhao C. Assessment of Image Quality in Augmented Reality Displays Using a Computational Model of Target Detectability [Electronic resource] / Chumin Zhao, Ryan Beams, Matthew Johnson, Aldo Badano // Presence: 2022 SID International Symposium Digest of Technical Papers. – 2022. – Volume 53, Issue 1. – P. 194–197. – DOI: https://doi.org/10.1002/sdtp.15451

# 5. МЕТОДИ ОЦІНЮВАННЯ ТА ПІДВИЩЕННЯ ЯКОСТІ ІНФОРМАЦІЙНИХ СИСТЕМ НА ПІДСТАВІ АНАЛІЗУ ДАНИХ

**Марк Ізраель[1,2], В. С. Харченко[2], О. О. Гордєєв[3]**

*[1]Holon Institute of Technology, Ізраїль*
*[2]Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»*
*[3]Луцький національний технічнний університет*

## 5.1. Вступ

### 5.1.1. Мотивація

Якість інформаційних систем або проектів (ІС – інформаційні системи) визначається за допомогою ряду моделей, які включають показники/фактори/характеристики якості та підлягають кількісній оцінці.

Користувачі відчули потребу у створенні моделі якості з метою кількісної та якісної оцінки якості програмного забезпечення. Відомі моделі якості, які є переважно ієрархічними моделями, засновані на критеріях якості та пов'язаних показниках. Усі такі моделі поділяються на три види відповідно до засобів, за допомогою яких ці моделі були сформовані. Перший вид — це теоретична модель, заснована на гіпотетичних співвідношеннях між змінними. Другий вид – це дані, які базуються на статистичному аналізі. Третій вид - це комбінована модель, у якій інтуїція використовується для визначення базового типу моделі, тоді як аналіз даних використовується для визначення констант моделі. Практично в більшості випадків адаптується комбінована модель [1].

Таким чином, тема оцінки якості інформаційних систем або проектів є багатогранною і стала об'єктом активного обговорення в літературі з різних точок зору.

Існує ряд популяцій для оцінки якості інформаційних систем або проектів. Популяції зазвичай обговорюються в літературі як найбільш детальні моделі.

Ці популяції відрізняються акцентами на різних аспектах ефективності інформаційних систем або проектів: ефективність для користувача, ефективність організації та прибутковість компанії.

Впровадження ГІС у процес картографування створило абсолютно новий тип користувача, який відрізняється від традиційного користувача карти. Ця ситуація вимагає ідентифікації спільних вимог для раціональної та точної реалізації таких систем [2]. Виявлення послуг на основі якості обслуговування (QoS), яке широко вивчається в традиційних веб-сервісах, також застосовується до геопросторових веб-сервісів [3]. Геопросторові веб-сервіси були активною сферою досліджень у контексті проблем геопросторової несумісності. Спільні зусилля промисловості та федеральних геопросторових клірингових центрів

були зосереджені на процесі стандартизації для пом'якшення проблем несумісності [3].

### 5.1.2. Поточний стан справ

У літературі ІС наводиться кілька визначень і показників успіху ІС. Як зазначено в [4], хоча існує майже стільки ж заходів, скільки й досліджень; очевидно, що остаточного визначення успіху ІС немає. Визначення успіху ІС може відрізнятися щодо різних типів ІС, які приносять різні переваги для окремих осіб, робочих груп і організацій [4].

Модель якості програмного забезпечення зазвичай визначається як набір характеристик і зв'язків між ними, що фактично забезпечує основу для визначення вимог якості та оцінки якості. За останні десятиліття було введено багато моделей якості програмного забезпечення (SWQM) [5].

Подібним чином зростає вплив розробки програмного забезпечення на основі компонентів (CBSD), зокрема, для розробки налаштовуваних, економічно ефективних, своєчасних і багаторазових великомасштабних і складних програмних систем. З метою побудови повного програмного рішення, основна увага приділяється створенню високоякісних частин і та їх подальшому з'єднанню. Одним із найважливіших процесів у CBSD є вибір програмного компонента відповідно до критеріїв кінцевого користувача. Модель якості відіграє важливу роль у процесі вибору компонентів [10].

Рух тотального управління якістю (TQM) використовує принципи розширення можливостей і цілей співробітників. TQM в першу чергу підкреслює необхідність зобов'язань керівництва. Керівництво, а також кожен працівник повинні бути обізнані про та прийняти політику якості, і від кожного очікується, що вони зобов'язуються дотримуватись якості, без опору. Процес розробки програмного забезпечення називається життєвим циклом. Від якості процесу залежить виробництво якісної продукції. Помилки на ранніх стадіях життєвого циклу, як правило, призводять до більшої кількості повторних робіт; такі помилки важче виправити, що, відповідно, вимагає ще більших витрат [11].

В останні роки дослідження в галузі систем якості та інформаційних проектів схилялися до висновку, що моделі повинні враховувати такі допоміжні параметри, як:

Сфера діяльності інформаційної системи або проекту, переваги користувачів тій саме області, які часто випливають з унікальності користувачів в конкретній області [12].

Кількісним показником ступеня, до якого елементи програмного забезпечення володіють даним атрибутом якості, є показник якості цього програмного забезпечення. Це обчислювальна функція, вхідними даними якої є дані програмного забезпечення (зазвичай вихідний код), а вихідним елементом є одне числове значення, яке можна інтерпретувати як ступінь, в межах якого в програмному забезпеченні присутній заданий атрибут якості. Отже, традиційні показники якості програмного забезпечення є еквівалентною кількісною мірою

атрибутів програмного забезпечення, які є основоположними для структурованого дизайну без урахування об'єктної орієнтації. Таким чином, показники програмного забезпечення можуть вимірювати якість шляхом ручного обчислення та перевірки вихідного коду, або шляхом використання еквівалентних автоматизованих інструментів [13].

Очевидно, що до питання методів моделювання для оцінки якості систем або інформаційних проектів необхідно включити додаткові аспекти. Наприклад, час вимірювання на різних етапах розробки, фактори якості продукту та процесу, набір показників якості, які можна виміряти на різних типах артефактів, таких як документ, модель і вихідний код, і, нарешті, конкретний механізм застосування динамічних ваг до факторів якості для визначення їх впливу на кінцеву якість продукту на основі сфери його застосування [14].

Хоча моделі успішності інформаційних систем (ІС) привернули велику увагу серед дослідників, існує загальний дефіцит досліджень, які проводяться для вимірювання успіху ГІС. У цьому документі пропонується модель успіху для вимірювання успіху ГІС шляхом розширення та модифікації попередніх моделей успіху ІС [6].

Дослідження включає оцінку ступеня різниці у важливості якісних характеристик у різних програмних проектах порівняно з важливістю якісних характеристик програмних проектів у сфері ГІС.

На основі огляду літератури можна визначити, що існуючі моделі не дають відповіді щодо оцінки якості проектів ІС і, зокрема, ГІС. Таке твердження має багато підстав - тема якості залежить від сфери організації, розміру організації, технологічного рівня співробітників, цілей організації, різних підходів до якості, а також спеціалізації співробітників в організації.

Якість інформаційних систем можна розглядати як лінійну комбінацію трьох аспектів: якості обслуговування, якості програмного забезпечення та якості інформації. Крім того, залежно від життєвого циклу інформаційних систем важливість зазначених аспектів змінюється. Аспект якості інформації у цій статті не розглядається.

Таким чином, метою даної статті є перевірка відповідності існуючих моделей якості загальним інформаційним системам або проектам (ІС), зокрема, ГІС-проектам [2]. Додатковою метою дослідження є запропонувати модель для оцінки якості. програмних проектів у сфері ГІС, які базуються на статистичній обробці даних.

## 5.2. Питання якості інформаційних систем або проєктів

Якість інформаційних систем описується за допомогою таких параметрів:

- «Якість – це сукупність властивостей і характеристик товару чи послуги, що забезпечують можливість відповідати необхідним вимогам...» [1];
- Якість є вимогою клієнтів, а не приписом інженерів, не комерційною потребою, не загальним чи юридичним приписом. Якість

базується на ознайомленні споживача з продуктом або послугою, виміряним відповідно до вимог до продуктів або вимог клієнта, у технічному, операційному чи суб'єктивному аспектах, і завжди представляє відповідну мету на конкурентному ринку.

Якість у сфері інформаційних систем або проектів можна розглядати як функцію, що складається з трьох показників, які впливають на успіх інформаційної системи компанії/організації:

- показник інформаційної системи/проекту;
- індикатор користувача;
- показник ефективності організації.

ДеЛон і Маклін [4] запропонували першу цілісну модель у 1992 році.

Завдяки широкому спектру досліджень було запропоновано модель успіху ІС. Ці моделі пропонують власне визначення успішності ІС та факторів, що впливають на успіх цієї ІС. Разом з цим було здійснено перше узагальнення багатьох точок зору на успішність ІС та її просування [4]. Ці спеціалісти розробили модель успіху ІС.

Ця модель привернула велику увагу дослідників ІС. Її можна вважати однією з популярних моделей, що має найбільший вплив у дослідженнях ІС [7], ДеЛон і Маклін [4] досягли значного прогресу, як показано на рис. 5.1. Вони провели всебічний пошук в літературних джерелах, що висвітлюють успіх ІС. Результати пошуку були опубліковані з 1981 по 1987 рік у семи випусках на тему ІС, та запропонували модель успіху ІС.


Рисунок 5.1 – Схема першої моделі ДеЛона і Макліна [4]

Після публікації моделі успіху ІС ДеЛона і Макліна було опубліковано кілька досліджень на цю тему з пропозицією контролювати або розширювати модель. Це призвело до більш повного розуміння моделі ІС.

В роботах, опублікованих в цій галузі, ця модель розглядалась з кількох точок зору. Модель включає контроль, коли стверджує, що успіх одного показника безпосередньо не призводить до успіху наступного показника.

Якби це було так, то для досягнення успіху всієї ІС необхідно було б лише забезпечити якість системи.

Відповідно до результатів опублікованих досліджень була запропонована розширена модель успіху ІС, представлена на рис. 5.2.

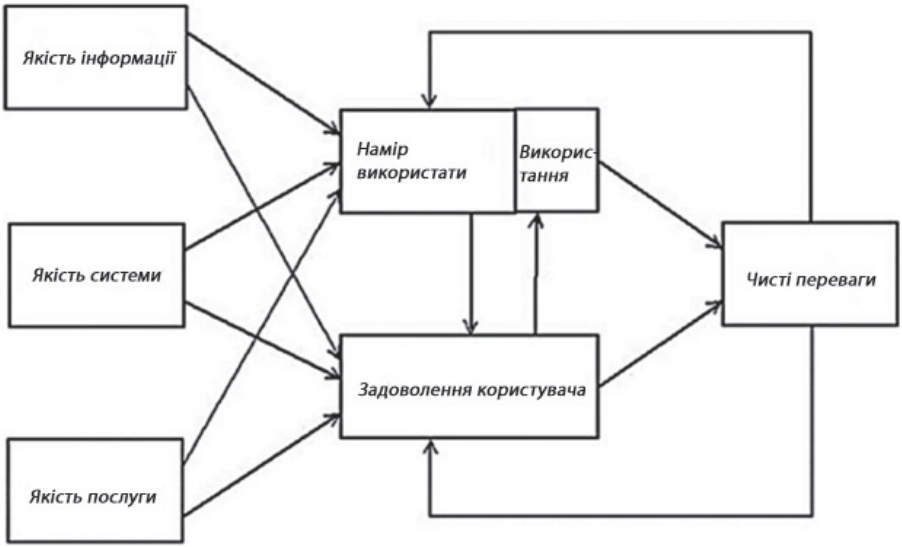



Рисунок 5.2 – Схема оновленої моделі ДеЛона і Макліна (2003) [4]

Три популярні аспекти відмінностей моделей можна визначити наступним чином:

1) додаванням якості обслуговування для відображення важливості обслуговування та підтримки успішних електронних продажів за допомогою систем;

2) додаванням наміру користувача при оцінці відношення користувача;

3) поєднанням особистого впливу та корпоративного впливу на структуру абсолютного вдосконалення;

Таксономія змінених категорій включала якість системи, інформації та обслуговування, наміри використання, використання, задоволення запитів користувача, та абсолютне вдосконалення.

Моделі успіху ІС ДеЛона і Макліна (1992, 2003) можуть бути використані як основа для вибору відповідних засобів ІС.

Дослідникам необхідно було обрати декілька відповідних засобів досягнення успіху відповідно до цілей і досліджуваних явищ, а також розглянути можливі взаємозв'язки між показниками успіху під час побудови моделі дослідження [6].

## 5.3. Географічні інформаційні системи (ГІС)

Географічні інформаційні системи (ГІС) – це програмні продукти (іноді в поєднанні з елементом обладнання), метою яких є створення, управління, обробка, аналіз і розповсюдження різноманітних даних, важливих з географічної точки зору.

Розвиток ГІС просувається в результаті зростаючого усвідомлення потенціалу використання ГІС. ГІС є відносно новою сферою загального користування, яка з часом розвивається. Це дозволяє обробляти дані для збільшення функціональності, і, як наслідок, зростає попит на системні дані.

Перевагою ГІС є спрощення та візуальне представлення типових явищ і поява ефективних альтернатив для тих, хто приймає рішення. ГІС-системи дозволяють передавати та отримувати дані багатьма методами, у тому числі високошвидкісною передачею даних [2].

У світлі вищевикладеного, моделі забезпечення якості, затверджені в області інформаційних систем, навряд чи будуть використовуватися для оцінки якості в області застосування ГІС.

Як зазначалося вище, географічні інформаційні системи (ГІС) є найбільш практичними для великого збору, зберігання, обробки, запитів і представлення даних у різних областях [3].

ГІС відрізняються від інших інформаційних систем або проектів, перш за все, географічним показником даних. Геолокація є дуже важливим показником, призначеним для спрощення обробки даних, класифікації великих обсягів даних, вимірювання даних і комплексного аналізу [2].

Сфера ГІС базується на п'яти критичних факторах [6].

Люди. Фахівці, що працюють з ГІС, характеризуються багатофункціональністю, яка виникає на основі необхідності поєднання даних різного змісту з затвердженими геоінформаційними системами. Загальна якість спеціалістів, які працюють з ГІС, полягає у знаннях географії та суміжних систем.

Дані. Дані в геоінформаційних системах характеризуються:

- різноманітністю джерел інформації, яка в більшості випадків надходить з території, з робочих кабінетів, баз даних;
- різноманітністю форм даних - формати, графічні дані, буквено-цифрові табличні дані.

Програмне забезпечення. Для нормалізації наявних даних необхідно збільшити графічний функціонал у поєднанні з класичним функціоналом обробки даних. Крім того, іноді потрібна обробка зображень і класифікація результатів.

Апаратне забезпечення. На етапі збору даних, призначених для використання в ГІС, використовуються спеціальні засоби, наприклад, антени RTK, засоби GPS, камери, датчики, кишенькові комп’ютери, мобільні телефони тощо.

Процедури/Методи. Етап, на якому розпізнається використання спеціальних методів ГІС, є етапом запиту даних. Використовуються розширені запити, наприклад, «покажіть об'єкти, включені до трикутника запиту», які зберігаються лише для використання ГІС.

### 5.4. Вивчення проблеми

Дослідження проводилися в Ізраїлі, серед фахівців, які працюють в різних сферах застосування інформаційних систем, включаючи ГІС. Населення включає системних операторів, спеціалістів з продуктивності системи та системних проектувальників у наступних областях (рис. 5.3):

- Мобільний додаток;
- Система управління документами;
- Інтернет-сайт і веб-додаток;
- BI та великі дані;
- Планування ресурсів підприємства та управління взаємовідносинами з клієнтами (ERP & CRM));
- Бібліотека карт та ГІС.

#### 5.4.1. Фактори та показники якості

Фактори якості, критерії та показники для систем або інформаційних проектів, які були розглянуті серед респондентів. Респондентам необхідно було оцінити рівень важливості факторів за шкалою Лікере 1-5.

#### 5.4.2. Результати дослідження

Дослідження проводили методом кількісної оцінки. Основною причиною вибору цього методу була перевірка кількісної відповідності між різними факторами та між якостями факторів. Достатня кількість респондентів заповнила опитування; таким чином, було отримано достатню кількість опитувальників, щоб зробити перші кількісні висновки на основі даних. Опитування проводилося планово, щоб уникнути неправильного розуміння визначення показників успіху.

Перевірку анкет проводили 5 експертів у галузі інформаційних систем.

На наступному етапі анкету було доопрацьовано відповідно до коментарів експертів.

Парадигми цього дослідження не були об'єднані, оскільки в літературі широко і всебічно розглядаються ІС. Тому не було потреби в попередньому якісному дослідженні перед визначенням факторів, які перш за все впливають на якість.

Для проведення дослідження в рамках даної роботи було відібрано 147 учасників, які розподілені за 7 сферами використання ІС (рис. 5.4).

Було проведено три основні методики статистичного аналізу (табл. 5.2):

- аналіз PCA був проведений для перевірки наступного припущення дослідження - Чи є різниця в розрахунках показників/критеріїв/характеристик якості між фахівцями, зайнятими у шести сферах використання інформаційних систем, та спеціалістами, зайнятими у сфері ГІС?
- аналіз MANOVA був проведений для перевірки наступного припущення дослідження - Чи є різниця в розрахунку показників/критеріїв/характеристик якості між спеціалістами, зайнятими у сфері ГІС, за спеціальностями, згаданими вище?
- аналіз EFA був проведений для побудови моделі оцінки якості, що є однією з цілей дослідження.

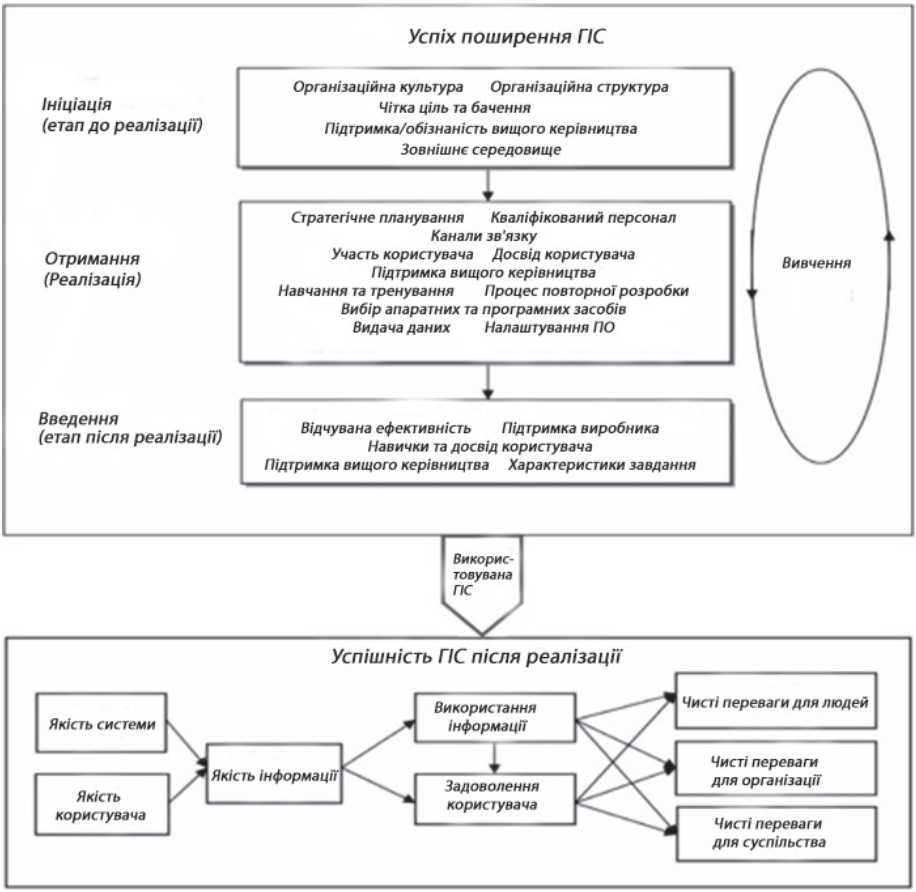



Рисунок 5.3 – Модель пропонована для оцінки успішності інформаційних проектів ГІС або програмних продуктів [6]

Таблиця 5.1 – Перелік факторів, показників та критеріїв якості

| Фактор якості | Опис фактора (підтверджено експертами) |
|---|---|
| Ефективність | Фактор якості, який використовується для оцінки здатності програмного продукту забезпечити належне використання з огляду на кількість ресурсів, доступних для продукту. |
| Гнучкість | Фактор якості, який використовується для оцінки зусиль, необхідних для внесення змін до робочого продукту/послуги. |
| Цілісність | Характеристика, яка використовується для оцінки доступу неавторизованих осіб до програми/послуги або даних. |
| Супроводжуваність | Фактор якості, який використовується для оцінки гнучкості програмного продукту/послуги до модифікацій і вдосконалень. |
| Продуктивність | Фактор якості, що використовується для оцінки впливу часових характеристик дії на продуктивність відповідного програмного продукту/послуги (відповідно до потреб користувача). |
| Надійність | Фактор якості, який використовується для оцінки здатності програмного продукту зберігати свою продуктивність на певному рівні при використанні в певних умовах. |
| Повторне використання | Фактор якості, який використовується для оцінки простоти забезпечення повторного використання програми або існуючих елементів програмного продукту/послуги. |
| Придатність до тестування | Фактор якості, який використовується для оцінки можливості усунення несправностей програмного продукту/послуги за допомогою попередньо визначеної процедури. |
| Придатність до модифікування | Критерій, який використовується для оцінки адаптованості програми до модифікацій або обробки відповідно до модифікацій операційного середовища та індивідуальних функціональних вимог. |
| Операційна сумісність | Критерій, який використовується для оцінки здатності програмного продукту/послуги до взаємодії з іншими системами в операційному середовищі користувача. |
| Коректність | Критерій, за яким оцінюється легкість усунення незначних дефектів, що виникають при переході з версії програмного продукту/послуги під час використання. |

Продовження табл. 5.1

| | |
|---|---|
| Точність | Показник, який використовується для оцінки ступеня, в межах якого програма відповідає вимогам користувача. |
| Доступність | Показник, який використовується для оцінки того, наскільки продукт/послуга практичні та доступні для користувача. |
| Придатність до змін | Показник, який використовується для оцінки зусиль щодо зміни робочого продукту/послуги. |
| Функціональність | Індикатор, який використовується для оцінки здатності програмного продукту належним чином функціонувати відповідно до попередньо встановлених продуктів. |
| Зрозумілість | Показник, який використовується для оцінки зрозумілості програмного продукту для користувача, відповідності програми вимогам і можливості її використання для вирішення різних завдань, які постають перед користувачем. |
| Здатність до використання | Показник, який використовується для оцінки доступності програмного продукту для навчання, функціональності та привабливості для користувача. |
| Видимість | Індикатор, який використовується для оцінки видимості програмного продукту/послуги. |
| Рівень безпеки | Показник, який використовується для оцінки інформаційної безпеки використання програмного продукту/послуги. |
| Стійкість | Показник, який використовується для оцінки здатності комп'ютерної системи протистояти помилкам під час роботи програми. |
| Здатність до розширення | Показник, який використовується для оцінки легкості внесення змін до роботи системи або елементів для розширення поточних можливостей програмного продукту/послуги. |
| Здатність до підтримки | Індикатор, який використовується для адаптації та прийнятності послуг в аспекті розрахунків та налаштування. Простота установки системи і локалізація несправностей. |
| Здатність до переходу | Показник, який використовується для оцінки здатності програмного продукту переходити з однієї версії обладнання або операційної системи на іншу. |
| Здатність до поєднання | Показник, який використовується для оцінки легкості підключення двох програмних продуктів/сервісів, що не заважає продуктивності. |
| Здатність до переміщення | Індикатор, який використовується для оцінки готовності програмного продукту для переходу з одного робочого середовища в інше. |


Рисунок 5.4 – Кругова діаграма розподілу в досліджуваній групі

У сфері ГІС 81 учасник заповнив опитування, 66 учасників завершили опитування в 6 інших областях. Цієї кількості недостатньо для перевірки припущень дослідження. Опитані учасники дослідження були відібрані шляхом особистого спілкування з дослідником. Дане дослідження було проведено серед фахівців в Ізраїлі. У цьому дослідженні досвід не усереднювався для дослідницьких цілей. 10 змінних, які отримали високу оцінку з 25 змінних і були відібрані відповідно до принципу 10 найкращих. Розрахунок багатофакторного дисперсійного аналізу за результатами Манова в області ГІС проводився лише для 10 змінних, зазначених вище, через обмеження розрахунків і кількість змінних порівняно з кількістю спостережень. Розрахунок результатів Manova вимагає мінімального співвідношення 1:3 кількості змінних порівняно з кількістю спостережень відповідно.

Таблиця 5.2 – Розподіл в досліджуваній групі

| Сфера застосування | Кількість респондентів | Процентний показник |
|---|---|---|
| Мобільний застосунок | 14 | 9.52% |
| Система управління документацією (DMS) | 12 | 8.16% |
| Інтернет-сайт веб-застосунок | 15 | 10.20% |
| BI та великі об'єми даних | 12 | 8.16% |
| ERP та CRM | 22 | 14.97% |
| Корпоративний портал та управління знаннями | 10 | 6.80% |
| Бібліотека ГІС та мап | 62 | 42.18% |
| Всього | 147 | 100% |

5.4.3. Обговорення результатів

5.4.3.1. Відмінність якісних характеристик в області досліджуваних систем

Було встановлено, що між різними ділянками існують значні відмінності параметрів.

Подібний висновок очікується, оскільки мова йде про різні важливі сфери, які відрізняються технологічним рівнем, цілями та призначенням програмного продукту/інформаційних продуктів, які відрізняються залежно від сфери.

На графіку (рис. 5.5) показано результати PCA для дисперсії 10 змінних, які найбільше підходять для аналізу.

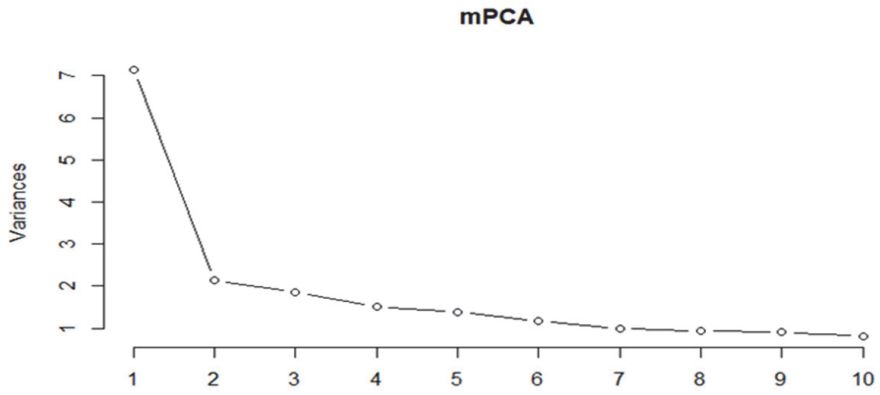



Рисунок 5.5 – Розподіл дисперсії серед 10 змінних з найбільш помітними відмінностями

На основі цього графіка можна виконати аналіз основних компонентів (PCA), використовуючи 2 основні фактори: PC1 і PC2. Слід взяти до уваги, що вертикальна вісь вказує на дисперсію між розподілом PCA. Це дозволяє переконатися, що існує велика різниця між значеннями PC1 і PC2 [15], тому ми можемо продовжити тестування на основі цих 2 факторів, як показано на рис. 5.6.

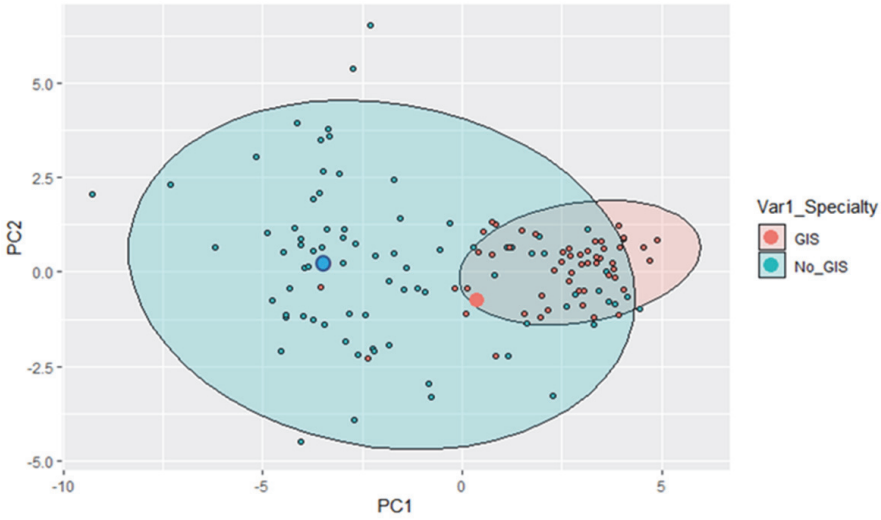



Рисунок 5.6 – Кластеризація результатів дослідження в розподілі ГІС порівняно з іншими областями з урахуванням «центрів тяжіння» двох груп

5.4.3.2. Відмінність характеристик якості у сфері ГІС

Відмінностей між різними функціями в області використання ГІС не виявлено, оскільки всі три функції належать до одних і тих саме параметрів «в цілому» і визначають їх важливість. Крім того, всі три групи однаково важливі для розробки ГІС, тому їх параметри не відрізняються одна від одної.
Це пояснюється використанням гнучкого методу як загальноприйнятого методу розробки програмних продуктів ГІС/інформаційних проектів.
Метод базується на глибокій зацікавленості кінцевих користувачів у процесі розробки, що значною мірою забезпечує уніфіковану адаптацію значущих факторів якості.

5.4.4. Пропонована модель оцінки якості інформаційних систем або проектів у сфері ГІС

Згідно з результатами FA було виявлено, що значення p було нижчим за 1% для даних ГІС та інших даних вибірки:

- перевірка гіпотези про достатність 3 факторів;
- хі-квадрат становить 486,33 на 228 ступенях свободи;
- p-значення 8,34e-21.

Крім того, для аналізу 25 параметрів між 3 змінними на основі аналізу була побудована ієрархічна дворівнева модель.

Ієрархічна дворівнева модель буде використана для дослідницького факторного аналізу – результати EFA.

На основі статистичної обробки даних, зібраних опитуванням операторів, спеціалістів з виконання програм та розробників ГІС, можна запропонувати модель оцінки якості інформаційних систем або проектів у сфері ГІС.

У цьому випадку необхідно узгодити час для вимірювання факторів успіху та встановити процес впровадження моделі.

На основі впливу факторів/показників/критеріїв якості, що використовуються як незалежні змінні, три залежні змінні можна співвіднести з трьома основними етапами розробки інформаційного проекту/програмного продукту, як показано на рис. 5.7:

- Етап до реалізації;
- Етап реалізації;
- Етап після реалізації.

Поділ програмного забезпечення на програмне забезпечення для розробки інформаційних проектів та програмне забезпечення для розробки програмних продуктів є загальним. Третій етап включає такі дії, як експлуатація, супроводження, обслуговування та підтримка тощо.

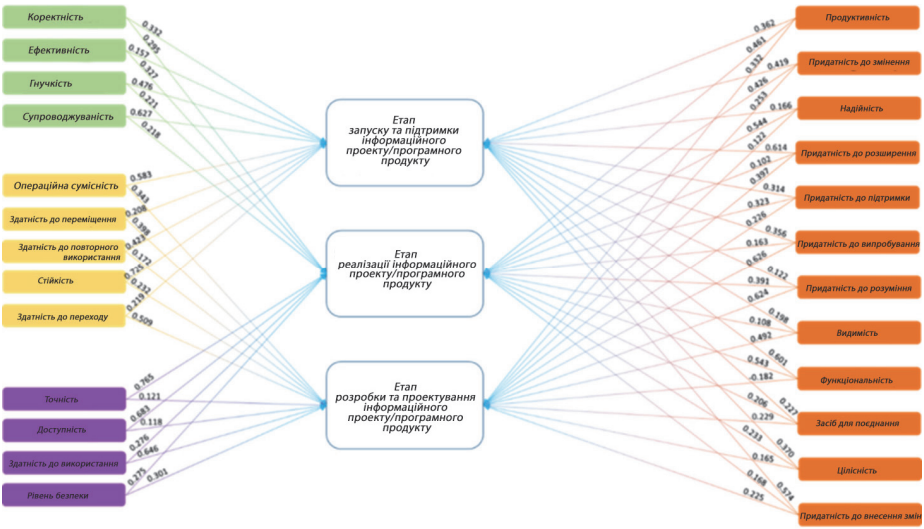



Рисунок 5.7 – Запропонована модель для оцінки якості інформаційних систем або проектів в сфері ГІС

Значення, які з’являються над стрілками, є ваговими коефіцієнтами, які представляють величину впливу фактора успіху на успіх інформаційної системи або інформаційного проекту в запропонованій моделі.

Для зручності модель можна представити у вигляді діаграми Венна (рис. 5.8).


Рисунок 5.8 – Діаграма Венна моделі, запропонованої для оцінки якості інформаційних систем або проектів у сфері ГІС з розподілом на трьох етапах життєвого циклу розробки інформаційної системи або проекту

## 5.5. Висновки

З метою поглибленого вивчення проблеми в подальші дослідження доцільно включити додаткові моменти, які можуть вплинути на методику оцінки якості інформаційних проектів/програмних продуктів у сфері ГІС.

Збільшення кількості учасників дослідження. Необхідно збільшити кількість учасників дослідження, наприклад, створити аналогічні передумови з точки зору кількісної оцінки, враховуючи важливість досвіду роботи учасників дослідження. Іншими словами, необхідно створювати дослідницькі когорти, схожі за розміром і досвідом роботи.

Розширення географії учасників дослідження. Наступне дослідження має бути проведено в кількох країнах із якомога різними культурами та технологіями. Це забезпечує міцну основу для запропонованої в дослідженні моделі оцінки якості інформаційних проектів ГІС.

Розширення ряду спеціальностей для роботи у сфері ГІС. Дослідження повинно поєднувати класифікацію спеціалістів із забезпечення якості в цій галузі.

Періодизація запропонованої в дослідженні моделі. Необхідно провести дослідження з метою періодизації запропонованої в цьому дослідженні моделі.

Збільшена кількість спостережень. Необхідно детально обґрунтувати результати дослідження, збільшити кількість спостережень перед відбором не менше ніж 75 спостережень на кожній ділянці. Розрахунок результатів Manova вимагає мінімального співвідношення 1:3 кількості змінних порівняно з кількістю спостережень відповідно.

Майбутні дослідження будуть присвячені розробці моделей якості для інформаційних систем або інформаційних проектів на основі сфери діяльності організації, користувачів інформаційних систем або інформаційних проектів і робочого середовища, в якому працює організація.

Література

1. С.К. Д'юбі, С. Гош, А. Рана, Порівняння моделей якості програмного забезпечення: Аналітичний підхід, Міжнародний журнал нової технології та методу проектування, Том. 2 (2), стор. 111-119, 2012.

2. Е. Тарантіно, Стандарти та якість у контексті ГІС, Робочий тиждень FIG 2003, Париж, Франція, 13-17 квітня 2003 р

3. М. Капріолі, Е.Тарантіно, Стандарти та якість в контекстах ГІС, Робочий тиждень FIG 2003 Париж, Франція, 13-17 квітня, 2003 р.

4. Г. Суббіах, А. Алам, Л. Хан, Б. Тураісінгха, Якості геопросторових даних в якості показників продуктивності веб-послуг, Засідання 15 Міжнародного симпозіуму щодо нових розробок в сфері геоінформаційних систем ACM GIS 2007, 2007.

5. В. Делоун, Е. Маклін, Перегляд успішності інформаційних систем, Засідання тридцять першого Міжнародного журналу електронної комерції Гавайїв, 45 Конференція системної науки (CD-ROM) , 2002.

6. О. Гордєєв, В. Кравченко, Н. Фоміних, В. Скляр, Еволюція моделей якості программного забезпечення в контексті Стандарту ISO 25010, Спрінгер-Верлаг Берлін Хайдельберг, adfa, стор. 1, 2014, DOI: 10.1007/978-3-319-07013-1_21.

7. К. А. Елдрандали, С. М. Нагуіб, М. М. Гассан, Модель вимірювання успіху географічних інформаційних систем, Журнал географічних інформаційних систем, стор. 328-347, 2015.

8. Н. Доаа, А. Мосад, Г.А. Хефни, Фактори якості мережевих застосунків: Огляд та пропонована концептуальна модель, Єгипетський журнал інформатики, том. 12, стор. 211-217, 2011.

9. О. Гордєєв, В. Кравченко, М. Фусані, Еволюція моделей якості программного забезпечення: придатність до використання, безпеки та незрілість, Останні досягнення в компютерній науці, стор. 519-523, 2015.

10. А. Равашдеш, М. Бассем, Нова модель якості програмного забезпечення для оцінки компонентів COTS, Журнал комп'ютерної науки, том 2 (4), стор. 373-381, 2006.

11. Н. Упадхяй, Б. Деспанде, В. Агравал, Вперед до моделі якості програмного забезпечення, Компютерна наука та інформаційні технології,

Засідання 1-ї Міжнародної конференції, Пенанг, Малайзія, 22-24 лютого 2011 р., Спрінгер, 2011, стор. 398-412. DOI: 10.1007/978-3-642-17857-3_40.

12. Е. Джордіадуі, GEQUAMO-A Типова, багаторівнева, налаштовувана модель якості програмного забезпечення, Журнал контролю якості програмного забезпечення, том 11 (4), стор. 313-323, 2003.

13. Д. Скапін, С. Люльєр, Ж. Вандердонкт, С. Маріяж, С. Бастієн, С. Фаренк, П. Паланк, Р. Бастід, Схема організації Основних положень щодо використання веб-ресурсів, суб'єктивних факторів та мережі: HFWeb, Засідання 7-ї Міжнародної конференції, Остін, Техас, США, 19 липня 2000 р. Спрінгер, 2000, стор. 1-23.

14. І. Ю. Мшелія, MECOT: Колекція засобів вимірювання якості програмного забезпечення, ЖУРНАЛ СИСТЕМНОЇ ІНТЕГРАЦІЇ 2019/1, 2019 р. 21-35, DOI: 10.20470/jsi.v10i1.360.

15. М. С. Хемаяті, Х. Рашіді, Моделі якості програмного забезпечення: Комплексний огляд та аналіз, Журнал новітніх досягнень в сфері електротехніки та комп'ютерних розробок, 2018, том 6, №. 1, стор. 59-76, DOI: 10.22061/JECEI.2019.1076.

16. С. П. Мішра, Ю. Саркар, С. Тарафдер, С. Датта, Д. П. Свейн, Р. Сайхом, С. Панда та М. Лаішрам, Багатоваріаційний аналіз статистичних даних – Аналіз основних компонентів (PCA), Міжнародний журнал досліджень в сфері тваринництва, том 7 (5), стор. 60-78, 2017, DOI 10.5455/ijlr.20170415115235.

# ЧАСТИНА ІІ. МЕТОДИ І ТЕХНОЛОГІЇ ОЦІНЮВАННЯ ТА ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

## 6. АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ОЦІНЮВАННЯ І ЗАБЕЗПЕЧЕННЯ КІБЕРБЕЗПЕКИ СИСТЕМ І СЕРВІСІВ ШТУЧНОГО ІНТЕЛЕКТУ

**О. С. Неретін, В. С. Харченко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

### 6.1. Вступ

Системи штучнуго інтелекту (СШІ) на сьогодні поширені у багатьох сферах, таких як державна, фінансова, медична, військова, побутова та інші сфери. Організація PricewaterhouseCoopers [1] прогнозує, що до 2030 року прискоренний розвиток систем штучного інтелекту може збільшити світовий ВВП на 14%.

Сучасні СШІ базуються на методах, які є вразливими до руйнівних атак, які є дуже небезпечних для їх нормального функціонування. У результаті, кіберзлочинці можуть контролювати ці системи та маніпулювати ними за власним бажанням.

Існує багато векторів атак на СШІ. Щоб зрозуміти ситуацію у цьому напрямку, необхідно класифікувати ці атаки та детально розглянути найважливіші з них. Виходячи з класифікації, атаки слід проаналізувати відповідно до рівня ризику всієї системи. На основі результатів аналізу, слідує визначити найшкідливіші атаки, які мають низький рівень контрзаходів від них.

Враховуючи поширеність СШІ в критичних сферах життя, важливо забезпечити їх захищеність, як одну з характеристик якості цих систем [2]. Це, в свою чергу, обумовлює актуальність удосконалення науково обґрунтованих методів і засобів об'єктивного оцінювання кіберзахисту СШІ.

Процес оцінки кібербезпеки СШІ є багатоетапним. Враховуючи те, що якість цього процесу залежить від повноти та достовірності інформації про вразливості і загрози системам, необхідно спочатку проаналізувати існуючі методи та засоби збирання та оброблення такої інформації. Слід підкреслити, що інформація про вразливості СШІ є погано впорядкованою та міститься в багатьох різних джерелах, таких як бази даних вразливостей, наукові статті, технічі звіти тощо. Тому важливо розробити модель та алгоритми для збору та аналізу даних про вразливості СШІ, щоб представити їх у відповідному зручному форматі.

Метою цього розділу є аналіз найбільш суттєвих загроз, вразливостей і контрзаходів для забезпечення кібербезпеки систем штучного інтелекту, а також аналіз існуючих методів та засобів збору інформації про вразливості СШІ.

Матеріал цього розділу базується на результатах, опублікованих в роботах [3-4].

## 6.2. Огляд атак на системи штучного інтелекту

Аналізувати кібербезпеку СШІ будемо шляхом аналізу основних вразливостей, атак на ці вразливості, наслідків цих атак та контрзаходів.

Вразливості СШІ здебільшого пов'язані з їх обмеженнями, якими доволі успішно користуються зловмисники. Звичайна ізоляційна стрічка може перетворити знак зупинки в зелений світ світлофора для системи розпізнавання безпілотного автомобіля [5], або навіть змусити його хибно думати, що треба пришвидшуватися. Одним із прикладів цього перетворення є результати дослідження команди з McAfee Advanced Threat Research, в якому вони примусили автопілот "Tesla" помилково пришвидшитися до 85 миль на годину, замість того, щоб триматися швидкості у 35 миль на годину [6]. На рисунку 6.1 зображені знаки обмеження швидкості, що використовувалися у експерименті. Чорна стрічка модифікувала цифру 3, зробивши її схожою на цифру 8 (рисунок з правої сторони), в результаті чого автопілот з штучним інтелектом зробив критичну помилку і почав пришвидшуватися, що могло б призвести до непередбачуваних наслідків.


Рисунок 6.1 – Звичайний та модифікований знаки обмеження швидкості (https://www.mcafee.com/blogs/other-blogs/mcafee-labs/model-hacking-adas-to-pave-safer-roads-for-autonomous-vehicles/)

Різноманітні вразливості СШІ надають хакерам можливість змушувати їх робити різні помилки, розкривати конфіденційну інформацію та навіть припиняти функціонування [7]. Вразливості цих систем можна розділити на дві основі групи - це "традиційні" вразливості програмного забезпечення (ПЗ) (атаки на інструментарій) та специфічні типи вразливостей, що можуть бути застосовані тільки для таких складних систем. Атаки на "традиційні" вразливості ПЗ складаються з двох основних типів [8]:

- класичні атаки на ПЗ - це атаки на відкрите програмне забезпечення, яке у дуже широко використовується в процесі розроблення СШІ та функціонування їх подальшого функціонування. Зловмисники можуть використовувати як вже відомі вразливості популярних продуктів зі світу СШІ, так і створювати нові шляхом додавання їх у ці продукти [9];
- типосквотінг (typosquatting) - це атаки, які досягають своїх цілей завдяки створенню бібліотек, назви яких схожі на назви популярних у отраслі інструментів.

Атаки на "традиційні" вразливості ПЗ спричиняють шкоду конфіденційності, цілісності та доступності СШІ.

Атаки на специфічні вразливості мають набагато більше різновидів. На рисунку 6.2 надано класифікацію цього типу атак [7, 10-12].

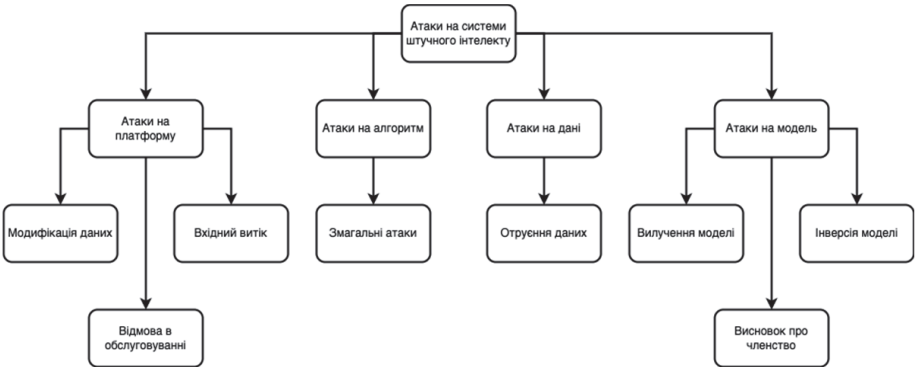



Рисунок 6.2 – Класифікація атак на системи штучного інтелекту

6.2.1. Атаки на платформу

"Атаки на платформу" (Platform attacks), на якій працюють СШІ [10], мають три наступних різновиди:

- модифікація даних (data modification) - це маніпулювання вхідними параметрами моделі. Досягається завдяки підбору таких вхідних даних, які зможуть вивести з ладу внутрішні механізми, які безпосередньо обробляють ці дані. Використовуючи атаки модифікації даних, зловмисники впливають на властивість цілісності СШІ;
- відмова в обслуговуванні (denial of service) - це уповільнення, або виведення з ладу СШІ. Виконується за допомогою надсилання дуже великого обсягу трафіку, який уповільнює доступ звичайним користувачам або повністю забороняє їм доступ. Атаки даного типу впливають на доступність системи;
- вхідний витік (input leakage) - це заволодіння вхідними даними користувачів. Відбувається за рахунок використання вразливостей в оточенні

СШІ, або завдяки її компрометації. Як наслідок, порушується конфіденційність даних СШІ.

### 6.2.2. Атаки на алгоритм

“Атаки на алгоритм” (Algorithm attacks), який використовується СШІ [10], мають основний різновид “Змагальні атаки на системи штучного інтелекту” (Adversarial attacks), які базуються на додаванні певного “шуму” до вхідних даних, в наслідок чого система робить хибне передбачення [13].

На рисунку 6.3 ілюструється один із прикладів цього типу атак. На ньому зображені дослідники із Бельгійського університету KU Leuven, що зламують відео аналітичний сервіс зі штучним інтелектом, використовуючи кольоровий роздрукований патерн [14-15]. Особа, що знаходиться ліворуч, правильно розпізнається системою як людина, а особу праворуч, система взагалі ніяк не класифікує, бо кольоровий роздрукований патерн заважає їй це зробити.

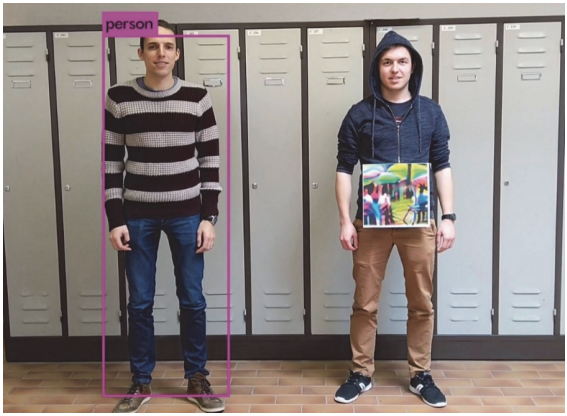



Рисунок 6.3 – Злам системи відео аналітичного сервісу (https://www.securityinfowatch.com/video-surveillance/video-analytics/article/21080107/researchers-hack-ai-video-analytics-with-color-printout)

За сприйняттям змагальні атаки можна розділити на наступні групи:

- видимі для людського ока зміни [5, 16-17] (на рисунку 6.4 шматочки ленти перетворюють знак “STOP” в зелене світло світлофора в “очах” СШІ [5]);
- невидимі для людського ока зміни, які стають причиною хибного результату класифікування моделлю СШІ [5, 18-21] (на рисунку 6.5 невидимі для людського ока зміни перетворюють зображення панди в мавпу в “очах” СШІ [5]).

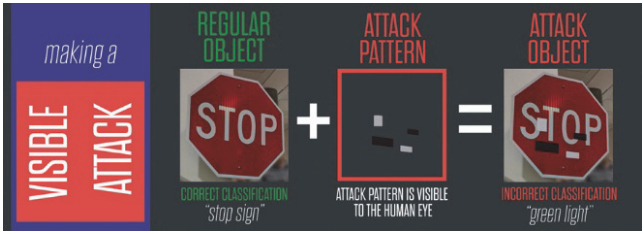



Рисунок 6.4 – Видимий для людського ока шаблон змагальної атаки (https://www.belfercenter.org/publication/AttackingAI)

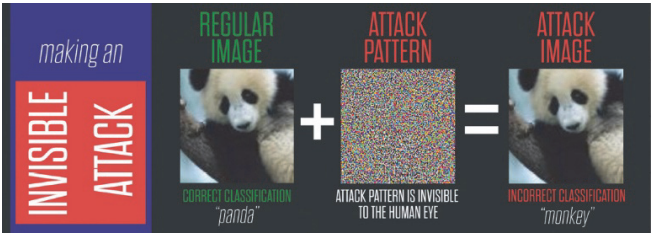



Рисунок 6.5 – Невидимий для людського ока шаблон змагальної атаки (https://www.belfercenter.org/publication/AttackingAI)

Використання змагальних атак на СШІ завдає шкоди цілісності цих систем.

### 6.2.3. Атаки на дані

“Атаки на дані” (Data attacks) [10], які використовуються СШІ у процессі навчання, мають основний різновиди “Атаки отруєння даних” (Data poisoning attacks). Ця вразливість виконується завдяки додаванню спеціально створених екземплярів даних, на яких навчаються СШІ. У більшій кількості випадків, це не заважає нормальній працездатності системи. Проте, передавання певних вхідних даних до СШІ, може змусити її видати хибний результат, саме на який і очікує зловмисник [22-29].

Використання зловмисниками атак цього типу порушує цілісність СШІ.

### 6.2.4. Атаки на модель

“Атаки на модель” (Model attacks), яку використовують СШІ [7], мають три наступних різновиди:

– вилучення моделі (model extraction) [7, 30] – це копіювання або викрадення моделі, яка підлягає атаці. Досягається шляхом методичного записування вхідних та вихідних даних моделі жертви. Даний різновид атак

створює два наступні ризики. По-перше, викрадення моделі надає зловмиснику її копію, розкриваючи інформацію про те, як працює система машинного навчання (МН). По-друге, викрадення моделі значно полегшує всі інші типи атак, які розглянуті в цій роботі. Розуміння того, як система працює, полегшує пошук шляхів, за допомогою яких цю систему можна зламати;

– висновок про членство (membership inference) [7, 10, 31-34] - це дізнавання подробиць про дані, на яких навчалася модель. Виконується шляхом вивчання вхідних та вихідних даних системи МН;

– інверсія моделі (model inversion) [7, 10, 35] - це визначення категорій вихідних даних моделі. Відбувається за рахунок внесення невеличких, поступових змін до вхідних даних для витягування певних даних, які є в моделі.

Використовуючи атаки на модель зловмисники мають змогу впливати на конфіденційність даних СШІ.

## 6.3. IMECA-аналіз кібератак і контрзаходів для забезпечення безпеки СШІ

Базуючись на вищезазначеній класифікації атак на СШІ, проведемо аналіз цих атак за рівнем небезпеки для систем у більш формальний спосіб, відповідно до основних положень методу Intrusion Modes Effects Criticality Analysis (IMECA) [36]. Для визначення рівня небезпеки проведемо аналіз за наступними параметрами:

– загроза – визначає те, за допомогою чого здійснюється атака на систему;

– вразливість – визначає слабку частину системи, завдяки якій можлива атака;

– атака – визначає тип вторгнення;

– наслідки – визначає наслідки, що можуть бути заподіяні атакою;

– ймовірність – визначає те, наскільки ймовірна поява атаки;

– тяжкість - визначає наскільки серйозною та небезпечною буде атака за наслідками;

– ризик – визначає сумарний вплив атаки на систему, що базується на ймовірності та тяжкості;

– контрзаходи – визначає заходи та дії, спрямовані на протидію атакам.

Комбінація показників ймовірності появи та тяжкості визначає рівень ризику – показника критичності. Ефективні контрзаходи, у свою чергу, можуть зменшити рівень цього показника. Високий рівень показника тяжкості з низьким рівнем контрзаходів мають ті атаки, що можуть спричинити найбільшу шкоду для систем штучного інтелекту. Результати IMECA аналізу наведено у таблиці 6.1. Він проведений з використанням даних з ресурсу [10] та є узагальненим з точки зору деталізації всіх елементів аналізу.

Таблиця 6.1 – IMECA-аналіз кібератак і контрзаходів для забезпечення безпеки СШІ

| # | Загроза | Вразливість | Атака | Наслідки | Критичність | | | Контрзаходи |
|---|---|---|---|---|---|---|---|---|
| | | | | | Ймовір-ність | Тяж-кість | Ризик | |
| 1 | Ретельно підібрані вхідні дані, які можуть модифікувати параметри СШІ | Сторонній код, недостатня перевірка вхідних даних | Модифікація даних | Втрата цілісності системи | Середня | Низька | Низький | Відмова від зайвого стороннього коду; ретельний контроль коду, який працює з вхідними даними |
| 2 | Масований за кількістю та обсягом потік вхідних даних | Відсутність фільтрації та обмеження кількості вхідних запитів | Відмова в обслуговуванні | Порушення доступності системи | Висока | Середня | Високий | Обмеження кількості запитів |
| 3 | Експлоїти до СШІ та їх оточення | Компрометація СШІ або її оточення, збереження даних у вихідному вигляді | Вхідний витік | Втрата даних, порушення конфіденційності | Висока | Низька | Середній | Шифрування даних |

Продовження табл. 6.1

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 4 | Дані, що подаються на вхід системи для аналізу | Вихідні обмеження СШІ (навчання базується на вивчені патернів), недостатній обсяг навчальних даних | Змагальні атаки | Втрата цілісності системи | Висока | Висока | Високий | Змагальні навчання, включення в дані змагальних прикладів, модифікація вхідних даних, захисна дистиляція, метод вилучення інваріантів, стискання функцій |
| 5 | Звичайні вхідні дані, які система навчена класифікувати невірно | Використання сторонніх даних, ненадійні дані, відсутня фільтрація даних | Отруєння даних | Втрата цілісності системи | Висока | Висока | Високий | Користуватися даними з надійних джерел, валідувати та фільтрувати дані, захищати їх під час використання |
| 6 | Запити до системи, що націлені на викраденя моделі | Принципи навчання СШІ | Вилучення моделі | Порушення конфіденційності | Висока | Середня | Високий | Обмеження кількості запитів |

Продовження табл. 6.1

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 7 | Запити | Принци- | Вис- | Втрата | Висока | Се- | Висо- | Обме- |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | до системи, що націлені на збір даних, на яких вона навчалася | пи навчання СШІ, залежність від зовнішніх даних | новок про членство | даних, порушення конфіденційності | | редня | кий | ження кількості запитів |
| 8 | Запити до системи, що націлені на витягування даних, на яких вона навчалася | Принципи навчання і функціонування СШІ | Інверсія моделі | Порушення конфіденційності | Висока | Середня | Високий | Обмеження кількості запитів |

За результатами аналізу атак за рівнем небезпеки для СШІ, побудуємо матрицю критичності (кібер ризиків) цих систем (таблиця 6.2) та матрицю критичності після впровадження контрзаходів (таблиця 6.3). Зелений колір позначає низький рівень ризику (атака 1), жовтий колір – середній рівень ризику (атака 3), червоний – високий рівень ризику (атаки 2, 4, 5, 6, 7, 8).

Таблиця 6.2 – Матриця критичності кібер ризиків СШІ

| | Тяжкість | | |
|---|---|---|---|
| Ймовірність появи | Низька | Середня | Висока |
| Низька | | | |
| Середня | 1 | | |
| Висока | 3 | 2, 6, 7, 8 | 4, 5 |

Таблиця 6.3 – Матриця критичності кібер ризиків СШІ після впровадження контрзаходів

| | Тяжкість | | |
|---|---|---|---|
| Ймовірність появи | Низька | Середня | Висока |
| Низька | | 2, 6, 7, 8 | |
| Середня | 1 | | |
| Висока | 3 | | 4, 5 |

На підставі аналізу матриці критичності (таблиця 6.3) атаки “Відмова в обслуговуванні” (2), “Вилучення моделі” (6), “Висновок про членство” (7) та “Інверсія моделі” (8) змінюють рівень ймовірності появи з високого на низький завдяки ефективним контрзаходам. Однак “Змагальні атаки” (4) та “Отруєння даних” (5) залишають системи в зоні високого ризику, бо наявні контрзаходи ніяк не толерують наслідки від цих атак.

## 6.4. Аналіз існуючих методів та засобів збору інформації про вразливості СШІ

Наразі не існує єдиного, комплексного методу для оцінки стану захисту систем штучного інтелекту від кібер злочинців. Кожна організація, яка використовує ці системи, вирішує питання оцінювання ризику самостійно, базуючись на власному досвіді або взагалі не робить таку оцінку. Ця ситуація може призвести до негативних наслідків, пов'язаних з ризиком втрати конфіденційності, цілісності та доступності ресурсів, що базуються на СШІ.

Сфера загальної кібербезпеки має у своєму арсеналі MITRE ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge) [37] - базу знань про тактики та прийоми противників, засновану на реальних спостереженнях. Частіше за все, ця база використовується як базова для розробки конкретних моделей і методологій загроз. Разом з цією загальною базою, корпорація MITRE розробила ATLAS (Adversarial Threat Landscape for Artificial-Intelligence Systems) [38] - базу знань про тактику противника, прийоми та тематичні дослідження для систем машинного навчання (МН). Тактики та техніки цієї бази доповнюють тактики та техніки бази ATT&CK. Таким чином, ці дві бази складають міцну основу для розроблення моделей, методів та засобів оцінювання загроз безпеки систем штучного інтелекту.

Деякі сфери, такі як авіація, займаються збором, зберіганням та аналізом реальних збоїв, що з ними коїлися. Це все робиться для того, щоб мати змогу детально аналізувати ці збої з ціллю запобігання їх повторній появі у майбутньому або пом’якшення їх негативного впливу. З тією ж самою ціллю була розроблена та активно підтримується у актуальному стані база даних

інцидентів штучного інтелекту [39]. Ця база є систематизованою колекцією інцидентів, які спричинили проблеми кібербезпеки СШІ.

Наукові джерела, що містять відкриті дані, такі як arXiv [40] та MDPI [41], містять багато різної інформації, що стосується СШІ та МН, їх класифікації, вразливостей та методів боротьби з ними. За допомогою поєднання цієї інформації з інформацією із бази даних інцидентів штучного інтелекту можна оцінити статистичний рівень ризику, який є первинною оцінкою ризиків напряму застосування конкретної СШІ.

Наряду з цим, можна виділити національну базу даних вразливостей NVD [42], яка є найбільш популярним сховищем стандартизованих даних. Ця база містить вразливості, кожна з яких має свій ідентифікатор загальних вразливостей та ризиків (CVE). На жаль, ця база даних не містить вразливостей безпосередньо для СШІ. Тому ці системи мають бути поділені на складові компоненти для того, щоб здійснювати ефективний пошук по цій базі даних. Надалі, користуючись сховищами коду, такими як GitHub [43], можливо отримати перелік вразливостей компонентів СШІ, за яким вже робити пошукові запроси до бази даних NVD.

У роботі [44] проводиться відображення вразливостей (CVE) програмного забезпечення на техніки ATT&CK за допомогою використання нейронної мережі. Таке саме відображення CVE's на техніки ATT&CK, але за допомогою мовної моделі, що базується на BERT (Bidirectional Encoder Representations from Transformers), надається у роботі [45]. Не зважаючи на те, що ці роботи обмежені лише цим відображенням, вони можуть бути корисними для аналізу вразливостей СШІ.

В роботі [46] з дослідження атак, захисту та інструментів СШІ та МН розроблена метамодель, яка складається з компонентів атак, методів та інструментів пом'якшення наслідків від цих атак. Ця модель наочно описує зв'язки понять за напрямом кібербезпеки систем ШІ, але не є повною.

Робота з дослідження оцінки загроз у машинному навчанні [47] є певним енциклопедичним документом, але їй не вистачає деталізації щодо СШІ як сервісів. Тому цю роботу можна розглядатися як дорожню карту у дослідженнях, пов'язаних з системами ШІ як сервісами.

У роботах [48-49], які присвячені моделюванню та аналізу загроз для СШІ, для оцінки безпеки цих систем пропонується застосування методології STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege). Також обговорюється застосування процесу FMEA (Failure Modes and Effects Analysis) для визначення того, як активи СШІ можуть бути порушені. Для оцінки серйозності загроз використовується метод DREAD (Damage Potential, Reproducibility, Exploitability, Affected Users and Discoverability), який був розроблений як доповнення до STRIDE. Увага цих робіт зосереджена на оцінці безпеки СШІ протягом їхнього життєвого циклу, проте використання цих систем як сервісів і безпекові показники цього випадку не розглядаються.

Виходячи з вищезазначеного аналалізу, можна зробити висновок про відсутність досконалих концептуальних моделей і методів збору інформації про вразливості СШІ для оцінювання їх кібербезпеки. В результаті чого СШІ проєктуються, розробляються та використовуються без грунтовного аналізу та забезпечення кібербезпеки.

Використовуючи досвід досліджень [44-49] і різноманітні дані про вразливості СШІ [37-43], важливо розробити моделі та засоби збору та аналізу цих вразливостей для забезпечення достовірної оцінки кібербезпеки цих систем.

## 6.5. Модель збору та аналізу вразливостей систем штучного інтелекту

Враховуючи попередній аналіз методів та засобів збору інформації про вразливості СШІ, розробимо контекстну діаграму рівня А0, яка являє собою функціональну модель процесу аналізу вразливостей цих систем. Результат розробки цієї моделі зображений на рисунку 6.6. Вхідні дані зображені стрілками зліва, вихідні – стрілками зправа, потрібні для виконання процесу механізми зображені стрілками знизу, а засоби контролю за цим процесом – стрілками зверху.

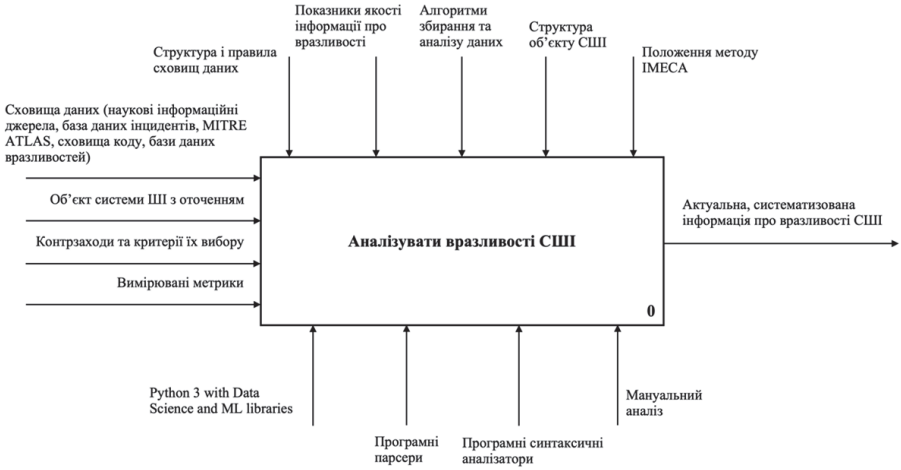



Рисунок 6.6 – Модель процесу аналізу вразливостей СШІ рівня А0

Процес аналізу вразливостей СШІ складається з наступних етапів:

1) Збір, нормалізація, фільтрація, об'єднання та перетворення даних з визначених сховищ з використанням засобів Big Data, що проводиться для отримання інформації про вразливості СШІ;

2) Поєднання даних зі сховищ з об'єктом СШІ, а саме вибір і представлення за визначеним форматом інформації про вразливості досліджуваної СШІ;

3) Оцінка критичності вразливостей СШІ з використанням різних методик, зокрема, IMECA.

Функційний опис процесів збору та аналізу вразливостей СШІ, який представлений IDEF (Integrated Definition) моделлю на зображенні вище, є першим кроком у подальшому розробленні досконалих моделей та методів оцінювання і забезпечення кібербезпеки систем штучного інтелекту як сервісів.

## 6.6. Загальні методи і рекомендації щодо забезпечення кібербезпеки СШІ, глобальна міждержавна взаємодія з питань безпечності СШІ

Інтерес урядів щодо безпеки систем штучного інтелекту значно зріс за останні роки [50-51]. Фахівці країн підкреслюють важливість прозорості, тестування та підзвітності алгоритмів цих систем та їх розробників [50]. Як приклад можно навести Комісію з національної безпеки штучного інтелекту (NSCAI) Сполучених Штатів Америки, яка наголосила на важливості створення гарантоздатних (надійних і безпечних) СШІ, які можна перевіряти за допомогою стандартизованої системи документації [52]. З цією метою комісія рекомендувала розробити стандарти для моделей ШІ, включаючи вимоги щодо того, які дані використовуються цими моделями, які параметри вони використовують та їхня вага, як вони навчаються і тестуються та які результати отримують. Це надасть змогу експертам виявляти вразливості технології СШІ, ризики потенційного маніпулювання з вхідними даними, а також інших неочікуваних результатів [50].

Регулятори також відіграють дуже важливу роль у цьому процесі. Вони можуть розробити механізми підзвітності та режими відповідальності для управління СШІ у разі виникнення кіберінцидентів. Це можуть бути базові вимоги до розробників СШІ щодо, наприклад, отримання сертифікатів, проходження аудитів та тестування з врахуванням специфічних характеристик штучного інтелекту. Розробники, які не виконують ці стандарти, створюючи СШІ, у випадку їх компрометації будуть нести відповідальність за заподіяну шкоду [50].

Стосовно захисту СШІ доцільно сформулювати такі основні рекомендації:

– по-перше, безпечний життєвий цикл розроблення СШІ є дуже важливим до впровадження [8, 53-54];

– "традиційні" типи вразливостей ПЗ можна пом'якшувати завдяки мінімізації використання стороннього коду чи ретельної перевірки захищеності того коду, без якого неможливо обійтися у даній конкретній ситуації;

– атакам на платформу можна протистояти, приділяючи більшу увагу процесу оброблення вхідніх даних, захищаючи платформу відомими,

стандартними механізмами боротьби з DoS атаками та контролюючи оточення системи щодо витоку вхідних даних користувачів [10];

– атакам на алгоритм (змагальним атакам) можна протистояти використовуючи навчання, яке включає у набір даних певні змагальні приклади [10, 55-56], а також завдяки модифікації вхідних даних [57]. Наряду з цим існують більш специфічні методи боротьби з цими атаками, такі, як наприклад, захисна дистиляція [58], метод вилучення інваріантів [59], стискання функцій [60] тощо;

– щоб протистояти атакам на дані, треба використовувати нескомпрометовані набори даних з надійних джерел, фільтрувати та валідувати ці дані, а також ретельно захищати їх під час використання. Дані мають бути не персоналізованими задля уникнення проблем конфіденційності. Крім того, завдяки лінійній регресії можна боротися з атаками даного типу [61]. Існують і більш специфічні методи для боротьби з цими атаками [28];

– атакам на модель можна протистояти завдяки обмеженню кількості запитів до системи.

## 6.7. Висновки

Проаналізувавши стан речей щодо кібербезпеки СШІ було виявлено, що вони вразливі як для класичних атак на ПЗ, так і для специфічних векторів атак, що притаманні тільки цим системам. Детальний аналіз класичних вразливостей ПЗ в процесі цього дослідження не здійснювався, оскільки цей напрям вже дуже добре опрацьований протягом багатьох років і не є унікальним для СШІ.

Визначено, що специфічні вектори атак на СШІ складаються з чотирьох основних груп: “Атаки на платформу”, “Атаки на алгоритм”, “Атаки на дані” та “Атаки на модель”.

“Атаки на платформу” за своєю сутністю дуже близькі до класичних атак на ПЗ. Модифікація даних, відмова в обслуговуванні, вхідний витік – це напрями, вже знайомі для спеціалістів з інформаційної безпеки (ІБ). Наразі існує багато методів боротьби з цими типами атак. За нашими оцінками тяжкість наслідків від цих атак знаходиться на рівні нижче середнього.

“Атаки на алгоритм”, а саме “Змагальні атаки” – це абсолютно новий тип атак, специфічний тільки для СШІ. Маючи високий рівень ймовірності появи, вони створюють велику загрозу цим системам та мають високий рівень тяжкості наслідків. Проте, основні контрзаходи щодо цих атак мають суто експериментальний характер. До цих контрзаходів можна віднести наступні: змагальне навчання; модифікація вхідних даних на предмет очищення від стороннього шуму тощо. Окрім того, існує певна кількість специфічних методів боротьби з цими загрозами, проте кожен з них охоплює досить вузький діапазон можливих атак. Все це ускладнює розроблення надійного захисту СШІ від даного типу атак.

“Атаки на дані”, а саме “Отруєння даних”, є наступним специфічним типом атак на СШІ. Цей тип атак має високий рівень ймовірності появи, що

суттєво збільшує їх наслідки для СШІ. Протистояння цим атакам мають лише рекомендаційний характер і полягають у використанні даних з надійних джерел, їх фільтрації та валідуванні, захисту даних під час навчання. На нашу думку, цих рекомендацій недостатньо для того, щоб мати СШІ, захищені від атак цього типу.

“Атаки на модель” є ще одним специфічним типом атак на СШІ. Для здійснення цього типу атак потрібно зробити велику кількість запитів до системи, що доволі легко контролюється і толерується шляхом обмеження кількості цих запитів (як це робиться для захисту від атак “Відмови в обслуговуванні”). Тому цей напрям має середній рівень тяжкості наслідків і не є пріоритетним для дослідження.

В результаті аналізу інформації стосовно вразливостей СШІ виявлено, що наразі відсутні досконалі методи збору цієї інформації та її подальшого аналізу. Визначено, що цей процес є багатокроковим і має забезпечувати коректність обробки, поєднання, аналізу та виділення корисної інформації з використанням інструментів великих даних для отримання актуальної та структурованої інформації про вразливості СШІ.

Важливим є висновок про необхідність в безпечному життєвому циклі розроблення СШІ. Для некритичних систем можуть бути менш жорсткі рекомендації та гнучкий життєвий цикл задля того, щоб не гальмувати їх застосування. Для критичних має бути доволі сувора та стандартизована система. Підкреслимо важливість прозорості, тестування, підзвітності алгоритмів та людей, які їх розробляють, і за певних умов, надають послуги з використанням СШІ. При відмовах у критичних СШІ розробники та операційники мають нести відповідальність за нанесену шкоду. Однак, ця рекомендація є доволі чутливою і вимагає детального відпрацювання.

СШІ створюють нові цінності для суспільства, тому для подальшого розвитку вкрай важливо підтримувати безпечність цієї технології як на етапі розроблення, так і на етапі експлуатації.

Література

1. PwC: The macroeconomic impact of artificial intelligence. (2018). [Online]. Available at: https://www.pwc.co.uk/economic-services/assets/macroeconomic-impact-of-ai-technical-report-feb-18.pdf.

2. Kharchenko, V., Fesenko, H., & Illiashenko, O. (2022). Basic model of non-functional characteristics for assessment of artificial intelligence quality [Online]. Available at: http://nti.khai.edu/ojs/index.php/reks/article/view/reks.2022.2.11.

3. Neretin, O., & Kharchenko, V. (2022). Ensurance of artificial intelligence systems cyber security: analysis of vulnerabilities, attacks and countermeasures. Journal of Lviv Polytechnic National University. Information Systems and Networks [Online]. Available at: https://science.lpnu.ua/uk/sisn/vsi-vypusky/vypusk-12-2022/zabezpechennya-kiberbezpeky-system-shtuchnogo-intelektu-analiz.

4. Neretin, O., & Kharchenko, V. (2022). Model for Describing Processes of AI Systems Vulnerabilities Collection and Analysis using Big Data Tools. In 2022 12th International Conference on Dependable Systems, Services and Technologies (DESSERT). DOI:10.1109/DESSERT58054.2022.10018811.

5. Comiter, M. (2019). Attacking Artificial Intelligence: AI's Security Vulnerability and What Policymakers Can Do About It. Belfer Center for Science and International Affairs, Harvard Kennedy School [Online]. Available at: https://www.belfercenter.org/sites/default/files/2019-08/AttackingAI/AttackingAI.pdf.

6. Povolny, S. (2020). Model Hacking ADAS to Pave Safer Roads for Autonomous Vehicles. McAfee Labs [Online]. Available at: https://www.mcafee.com/blogs/other-blogs/mcafee-labs/model-hacking-adas-to-pave-safer-roads-for-autonomous-vehicles/.

7. Lohn, A. (2020). Hacking AI. Center for Security and Emerging Technology. DOI:10.51593/2020CA006.

8. Lohn, A. (2021). Poison in the Well. Center for Security and Emerging Technology. DOI:10.51593/2020CA013.

9. Ruef, M. (2020). Hacking Artificial Intelligence - Influencing and Cases of Manipulation [Online]. Available at: https://www.researchgate.net/publication/338764153_Hacking_Artificial_Intelligence_-_Influencing_and_Cases_of_Manipulation.

10. Kim, A. (2020). The Impact of Platform Vulnerabilities in AI Systems. Massachusetts Institute of Technology [Online]. Available at: https://dspace.mit.edu/bitstream/handle/1721.1/129159/1227275868-MIT.pdf.

11. Hartmann, K., & Steup, C. (2020). Hacking the AI - the Next Generation of Hijacked Systems. In 12 International Conference on Cyber Conflict (CyCon). DOI:10.23919/CyCon49761.2020.9131724.

12. Bursztein, E. (2018). Attacks against machine learning — an overview. Personal Site and Blog featuresing blog posts publications and talks [Online]. Available at: https://elie.net/blog/ai/attacks-against-machine-learning-an-overview/.

13. Wang, Y., Sun, T., Li, S., Yuan, X., Ni, W., Hossain, E., Poor, H. V. (2023). arXiv preprint arXiv:2303.06302. DOI:10.48550/arXiv.2303.06302.

14. Griffin, J. (2019). Researchers hack AI video analytics with color printout [Online]. Available at: https://www.securityinfowatch.com/video-surveillance/video-analytics/article/21080107/researchers-hack-ai-video-analytics-with-color-printout.

15. Thys, S., Ranst, W.V., & Goedemé, T. (2019). Fooling automated surveillance cameras: adversarial patches to attack person detection. arXiv preprint arXiv:1904.08653. DOI:10.48550/arXiv.1904.08653.

16. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., & Song, D. (2018). Robust Physical-World Attacks on Deep Learning Models. arXiv preprint arXiv:1707.08945. DOI:10.48550/arXiv.1707.08945.

17. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Tramer, F., Prakash, A., Kohno, T., & Song, D. (2018). Physical Adversarial Examples for

Object Detectors. arXiv preprint arXiv:1807.07769. DOI:10.48550/arXiv.1807.07769.

18. Su, J., Vargas, D.V., & Sakurai, K. (2019). Attacking convolutional neural network using differential evolution. IPSJ Transactions on Computer Vision and Applications. DOI:10.1186/s41074-019-0053-3.

19. Goodfellow, I.J., Shlens, J., & Szegedy, C. (2015). Explaining and Harnessing Adversarial Examples. arXiv preprint arXiv:1412.6572. DOI:10.48550/arXiv.1412.6572.

20. Papernot, N., McDaniel, P., & Goodfellow, I.J. (2016). Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples. arXiv preprint arXiv:1605.07277. DOI:10.48550/arXiv.1605.07277.

21. Catak, F.O., & Yayilgan, S.Y. (2021). Deep Neural Network based Malicious Network Activity Detection Under Adversarial Machine Learning Attacks. In International Conference on Intelligent Technologies and Applications, 280-291. DOI:10.1007/978-3-030-71711-7_23.

22. Volborth, M. (2019). Detecting backdoor attacks on artificial neural networks [Online]. Available at: https://ece.duke.edu/about/news/detecting-backdoor-attacks-artificial-neural-networks.

23. Vincent, J. (2020). Twitter taught Microsoft's AI chatbot to be a racist asshole in less than a day. The Verge [Online]. Available at: https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist.

24. Ji, Y., Liu, Z., Hu, X., Wang, P., & Zhang, Y. (2019). Programmable Neural Network Trojan for Pre-Trained Feature Extractor. arXiv preprint arXiv:1901.07766. DOI:10.48550/arXiv.1901.07766.

25. Yang, Z., Iyer, N., Reimann, J., & Virani, N. (2019). Design of intentional backdoors in sequential models. arXiv preprint arXiv:1902.09972. DOI:10.48550/arXiv.1902.09972.

26. Zhang, J., Song, B., Han, B., Liu, L., Niu, G., & Sugiyama, M. (2023). Assessing Vulnerabilities of Adversarial Learning Algorithm through Poisoning Attacks. arXiv preprint arXiv:2305.00399. DOI:10.48550/arXiv.2305.00399.

27. Pal, S., Wang, R., Yao, Y., & Liu, S. (2023). Towards Understanding How Self-training Tolerates Data Backdoor Poisoning. arXiv preprint arXiv:2301.08751. DOI:10.48550/arXiv.2301.08751.

28. Jagielski, M., Oprea, A., Biggio, B., Liu, C., NitaRotaru, C., & Li, B. (2018). Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In 2018 IEEE Symposium on Security and Privacy (SP), 19–35. DOI:10.1109/SP.2018.00057.

29. Aryal, K., Gupta, M., & Abdelsalam, M. (2023). Analysis of Label-Flip Poisoning Attack on Machine Learning Based Malware Detector. arXiv preprint arXiv: 2301.01044. DOI:10.48550/arXiv.2301.01044.

30. Karmakar, P., & Basu, D. (2023). Marich: A Query-efficient Distributionally Equivalent Model Extraction Attack using Public Data. arXiv preprint arXiv: 2302.08466. DOI:10.48550/arXiv.2302.08466.

31. Matsumoto, T., Miura, T., & Yanai, N. (2023). Membership Inference Attacks against Diffusion Models. arXiv preprint arXiv: 2302.03262. DOI:10.48550/arXiv.2302.03262.

32. Salem, A., Zhang, Y., Humbert, M., Berrang, P., Fritz, M., & Backes, M. (2018). ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. arXiv preprint arXiv:1806.01246. DOI:10.48550/arXiv.1806.01246.

33. Rahman, A., Rahman, T., Lagani`ere, R., Mohammed, N., & Wang, Y. (2018). Membership Inference Attack against Differentially Private Deep Learning Model [Online]. Available at: https://www.tdp.cat/issues16/tdp.a289a17.pdf.

34. Rezaei, S., & Liu, X. (2022). On the Discredibility of Membership Inference Attacks. arXiv preprint arXiv:2212.02701. DOI:10.48550/arXiv.2212.02701.

35. Nguyen, N-B., Chandrasegaran, K., Abdollahzadeh, M., & Cheung N-M. (2023). Re-thinking Model Inversion Attacks Against Deep Neural Networks. arXiv preprint arXiv: 2304.01669. DOI:10.48550/arXiv.2304.01669.

36. Babeshko, I., Illiashenko, O., Kharchenko, V., & Leontiev, K. (2022). Towards Trustworthy Safety Assessment by Providing Expert and Tool-Based XMECA Techniques. Mathematics 2022, 10, 2297 [Online]. Available at: https://doi.org/10.3390/math10132297.

37. MITRE corporation. ATT&CK framework [Online]. Available at: https://attack.mitre.org/.

38. MITRE corporation. ATLAS framework [Online]. Available at: https://atlas.mitre.org/.

39. McGregor, S. (2020). Preventing Repeated Real World AI Failures by Cataloging Incidents: The AI Incident Database [Online]. Available at: https://arxiv.org/abs/2011.08512.

40. arXiv.org e-Print archive [Online]. Available at: https://arxiv.org/.

41. MDPI - Publisher of Open Access Journals [Online]. Available at: https://www.mdpi.com/.

42. The national vulnerability database (NVD) [Online]. Available at: https://nvd.nist.gov/.

43. GitHub [Online]. Available at: https://github.com/.

44. Aditya Kuppa, Lamine Aouad, & Nhien-An Le-Khac. 2021. Linking CVE's to MITRE ATT&CK Techniques. In The 16th International Conference on Availability, Reliability and Security (ARES 2021), August 17–20, 2021, Vienna, Austria. ACM, New York, NY, USA, 12 pages. DOI:10.1145/3465481.3465758.

45. Grigorescu, O., Nica, A., Dascalu, M., & Rughinis, R. (2022). CVE2ATT&CK: BERT-Based Mapping of CVEs to MITRE ATT&CK Techniques [Online]. Available at: https://www.mdpi.com/1999-4893/15/9/314.

46. Fazelnia, M., Khokhlov, I., & Mirakhorli, M. (2022). Attacks, Defenses, And Tools: A Framework To Facilitate Robust AI/ML Systems [Online]. Available at: https://arxiv.org/abs/2202.09465.

47. Tidjon, L.N., & Khomh, F. (2022). Threat Assessment in Machine Learning based Systems [Online]. Available at: https://arxiv.org/abs/2207.00091.

48. Mauri, L., & Damiani, E. (2022). Modeling Threats to AI-ML Systems Using STRIDE [Online]. Available at: https://www.mdpi.com/1424-8220/22/17/6662.

49. Wilhjelm, C., & Mussa, Y. (2020) A Threat Analysis Methodology for Security Requirements Elicitation in Machine Learning Based Systems. IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C) [Online]. Available at: https://ieeexplore.ieee.org/document/9282660.

50. Wolff, J. (2020). How to improve cybersecurity for artificial intelligence. The Brookings Institution [Online]. Available at: https://www.brookings.edu/research/how-to-improve-cybersecurity-for-artificial-intelligence/.

51. Newman, J.C. (2019). Toward AI Security GLOBAL ASPIRATIONS FOR A MORE RESILIENT FUTURE [Online]. Available at: https://cltc.berkeley.edu/wp-content/uploads/2019/02/Toward_AI_Security.pdf.

52. National Security Commission on Artificial Intelligence. First Quarter Recommendations. (2020) [Online]. Available at: https://drive.google.com/file/d/1wkPh8Gb5drBrKBg6OhGu5oNaTEERbKss/view.

53. Pupillo, L., Fantin, S., Ferreira, A., & Polito, C. (2021). Artificial Intelligence and Cybersecurity. CEPS Task Force Report [Online]. Available at: https://www.ceps.eu/wp-content/uploads/2021/05/CEPS-TFR-Artificial-Intelligence-and-Cybersecurity.pdf.

54. Neustadter, D. (2020). Why AI Needs Security. Synopsys Technical Bulletin [Online]. Available at: https://www.synopsys.com/designware-ip/technical-bulletin/why-ai-needs-security-dwtb-q318.html.

55. Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., & McDaniel, P. (2020). Ensemble Adversarial Training: Attacks and Defenses. arXiv preprint arXiv:1705.07204. DOI:10.48550/arXiv.1705.07204.

56. Yuan, X., He, P., Zhu, Q., & Li, X. (2018). Adversarial Examples: Attacks and Defenses for Deep Learning. arXiv preprint arXiv:1712.07107. DOI:10.48550/arXiv.1712.07107.

57. Dziugaite, G.K., Ghahramani, Z., & Roy, D.M. (2016). A study of the effect of JPG compression on adversarial images. arXiv preprint arXiv:1608.00853. DOI:10.48550/arXiv.1608.00853.

58. Papernot, N., McDaniel, P., Wu, X., Jha, S., & Swami, A. (2016). Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks. In 2016 IEEE Symposium on Security and Privacy (SP), 582-597. DOI:10.1109/SP.2016.41.

59. Ma, S., Liu, Y., Tao, G., Lee, W.C., & Zhang, X. (2019). NIC: Detecting Adversarial Samples with Neural Network Invariant Checking. In NDSS [Online]. Available at: https://www.ndss-symposium.org/ndss-paper/nic-detecting-adversarial-samples-with-neural-network-invariant-checking/.

60. Xu, W., Evans, D., & Qi, Y. (2018). Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In Network and Distributed Systems Security Symposium (NDSS). DOI:10.14722/ndss.2018.23198.

61. Liu, C., Li, B., Vorobeychik, Y., & Oprea, A. (2017). Robust linear regression against training data poisoning. In Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, 91–102. DOI:10.1145/3128572.3140447.

# 7. АНАЛІТИЧНІ ТА ЕКСПЕРИМЕНТАЛЬНІ МЕТОДИ ОЦІНЮВАННЯ ФУНКЦІЙНОЇ ТА КІБЕРБЕЗПЕКИ РОБОТОТЕХНІЧНИХ СИСТЕМ

**А. І. Абакумов, В. С. Харченко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 7.1. Вступ

Робототехнічні системи (РТС) щоденно застосовуються з метою підвищення ефективності, швидкості та точності виконання виробничих процесів. Зростаючий обсяг використання РТС зобов'язує виробників пріоритезувати питання функційної та кібербезпеки під час розробки, розгортання та експлуатації цих систем. РТС можуть спричинити серйозні травми працівникам у разі виникнення несправності, неочікуваної поведінки та/або неправильного використання. Крім ризиків безпеки, кібербезпека також є критичним питанням РТС, особливо у випадках, коли компоненти цієї системи підключені до єдиної мережі. РТС мають бути безпечні та добре захищені. Якщо ні, вони можуть стати небезпечними інструментами здатними сіяти хаос та завдати суттєвої шкоди своєму оточенню та безпосередньо людям, яким вони надають послуги [1]. Людство вже зіштовхнулось з деякими наслідками серйозних проблем із кібербезпекою пристроїв інтернету речей, що завдали шкоди компаніям та бізнесам, а також окремим користувачам. Проблеми з кібербезпекою роботів можуть мати значно більший вплив. Відомі інциденти, пов’язані з роботами:

- робот-охоронець у Стенфордському торговому центрі в Силіконовій долині збив малюка; на щастя, дитина серйозно не постраждала; [2]
- робот китайського виробництва потрапив в аварію на технічному ярмарку в Шеньчжені, розбивши скло вітрини та поранивши людину, що знаходилась поруч; [3]
- у 2007 році роботизована гармата вбила дев'ятьох солдатів і серйозно поранила ще 14 під час стрільб через несправність цього робота; [4]
- згідно з нещодавнім дослідженням, роботизована хірургія пов’язана з 144 смертями в США. [5]

Не дивлячись на те, що цей перелік інцидентів здебільшого складається з нещасних випадків, вони наочно демонструють, наскільки небезпечними можуть бути скомпрометовані або зламані РТС. Незважаючи на очевидну необхідність забезпечення безпеки та кіберзахисту РТС, гетерогенність цих систем, їх велику кількість і різноманіття, а також обмеженість ресурсів перешкоджають впровадженню дієвих механізмів функційної та кібербезпеки.

Належним чином сплановані та відповідно виконані заходи допоможуть забезпечити безпеку та кібербезпеку РТС. Оцінка готовності РТС до

черезвичайних ситуацій та кібератак, а також тестування безпеки та кіберзахисту повинні проводитися на кожному етапі життєвого циклу РТС, від проектування та розробки до впровадження та експлуатації. Така оцінка може включати аналіз потенційних загроз, визначення критично важливих компонентів і процесів, які необхідно захистити, а також визначення стратегій виявлення та відновлення РТС після інциденту.

З метою покращення показників тестового покриття, точності та достовірності результатів, часу виконання та вартості процесів оцінювання безпеки та кібербезпеки РТС пропонується використовувати поєднання різних аналітичних та експериментальних методів, серед яких виділяють наступні:

- Тестування на проникнення (ТнП);
- IMECA (Intrusion Modes and Effect Criticality Analysis);
- Оцінка ризиків та вразливостей (Risks & vulnerabilities assessment (R&VA));
- Аналіз дерева атак (Attack Tree Analysis (ATA));
- Ін'єктування несправностей/вразливостей (Faults and Variabilities Injection Testing (F&VIT));
- STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privileges);
- Блок-схеми надійності, захищеності та безпеки (Reliability (Safety) Block Diagrams (R(S)BD)).

Нажаль, на сьогоднішній день не існує стандартизованої методології оцінювання функційної та кібербезпеки адаптованої під вимоги РТС. Однак, на підставі аналізу існуючих джерел інформації, що базуються на практичному досвіді авторів у сфері забезпечення функційної та кібербезпеки РТС, можна сформувати низку комбінацій запропонованих методів та їх етапів з урахуванням специфіки архітектури та особливостей РТС, а також можливих загроз безпеці та кібербезпеці на кожному з рівнів їх (систем) розробки, розгортання та використання.

Об'єктом дослідження є процеси проведення функційної та кібербпезпеки РТС. Предметом дослідження є методи і засоби оцінювання безпеки та кібербезпеки РТС на різних етапах їх розроблення і використання. Метою дослідження є підвищення точності та достовірності, а також зменшення часу та вартості виконання оцінювання функційної та кібербезпеки РТС.

Для досягнення поставленої мети сформульовані наступні задачі дослідження:

- розглянути архітектуру сучасних РТС, що використовують в промислових умовах як об'єкту оцінювання;
- зібрати та проаналізувати існуючи загрози функційної та кібербезпеки РТС;
- проаналізувати сучасні методи та засоби оцінювання функційної та кібербезпеки та кожен з їх етапів;
- розробити комбінації методів оцінювання функційної та кібербезпеки, враховуючи їхні сильні та слабкі сторони, оцінити загальний

вплив на такі показники оцінювання, як точність та достовірність, час виконання та вартість;

- обрати найкращу з розглянутих комбінацій методів оцінювання функційної та кібербезпеки РТС на підставі експертного аналізу.

## 7.2. Архітектура РТС як об'єкту оцінювання

Автори [6] визначають РТС, як системи з числовим програмним управлінням, якими можна керувати автоматично, і які є перепрограмованими багатоцільовими маніпуляторами, що можуть рухатися по трьох або більше осях. Деякі роботи запрограмовані на виконання певних дій, які повторюються без змін і з високим ступенем точності.

За словами авторів [7] РТС, що використовуються у виробництві переважно у процесах зварювання, збірки кузовів автомобілів, покраски, автоматизації переміщення матеріалів, механічної обробки, а також для кооперації у форматі людина-робот для виконання завдань.

У звіті World Robotics 2021 [8] зазначається, що класифікація на промислових роботів та сервісних роботів здійснюється відповідно до їхнього цільового призначення. Промислові роботи - це роботи «для використання в додатках промислової автоматизації», тоді як сервісний робот «виконує корисні завдання для людей або обладнання, за винятком додатків промислової автоматизації». Правда полягає в тому, що в промисловості існують сценарії, коли роботам і людям доводиться ділити простір і завдання, а отже, промислові роботи більше не обмежуються безпечною зоною. Все частіше ми зустрічаємо сервісних роботів, які за визначенням працюють у промислових галузях. Фактично, у так званій Індустрії 4.0 роботи, ключову роль відіграють саме колаборативні роботи (коботи). Сьогодні ми б не говорили про колаборативну робототехніку без попереднього розвитку систем промислових роботів та їхнього руху до інтелектуальних рішень автоматизації, заснованих на взаємодії з людиною.

Механічно кобот – це маніпулятор з двома або більше шарнірами, що закінчується кінцевим ефектором (наприклад, плоскогубці, різак, лазерний зварювальний апарат), що взаємодіє з навколишнім середовищем. [9] Коботи працюють відповідно до програми завдань. [10] Такі роботи підключаються до своєї мережі через контролер, який може являти собою «коробку» з набором пов'язаних між собою комп'ютерів та інших модулів (привідний блок, контакторний блок, панель керування тощо), що керує роботом і підсистемою взаємодії людини з роботом (наприклад, джойстиками, перемикачами, індикаторами стану тощо). Система управління контролює та запускає механічні частини кобота, а також керує процесом взаємодії між ним, навколишнім середовищем, оператором та сервісами кобота (наприклад, веб-додаток, хмарні сервіси тощо). Контролер може працювати в автоматичному та/або ручному режимі. В автоматичному режимі він призначений для регулярної роботи робота на виробництві. Коли кобот працює в ручному

режимі, оператор може взаємодіяти з ним за допомогою пристрою, що називається навчальним планшетом, щоб приводити в дію рухи кобота при виконанні поставленої перед ним програми завдань. Зазвичай контролер та навчальний планшет мають доступ комплекту для розробки ПЗ (SDK) з метою програмування та перепрограмування. Цей процес зазвичай здійснюється в сервісній підмережі, а безпосередня взаємодія між коботом і контролером відбувається в окремій (заводській) підмережі. Приклад архітектури кобота наведено на рисунку 7.1.

Такі РТС, як колаборативні роботи, були розроблені для використання на промислових об'єктах з обмеженим доступом до них та декількома вбудованими механізмами безпеки (наприклад, фізичне блокування блоку управління, контрольна сума модифікації налаштувань безпеки, захист інтерфейсу взаємодіє людини з роботом паролем, зашифрований механізм оновлення ПЗ та ін. [11]). Але все ж таки є декілька точок входу (обведені пунктирним овалом на рис. 7.1), що зазвичай є відкритими в сучасних РТС, які хакери можуть використати в зловмисних цілях. Отже, до основних компонентів кобота можна виділити наступні:

- програму завдань (ПЗ);
- мережу (М);
- зовнішні інтерфейси (ЗІ);
- комплект для розробки (КДР);
- контролер (К), що включає:
  - файлову систему (ФС);
  - прошивку (П);
  - систему контролю доступу (СКД).


Рисунок 7.1 – Архітектура РТС (на прикладі кобота)

## 7.3. Загрози безпеки РТС

Незважаючи на те, що РТС надають безліч переваг, вони також можуть становити потенційні ризики безпеки та кібербезпеки. Функційна безпека є критичним аспектом використання РТС, а забезпечення їх безпечної роботи є важливим для запобігання нещасних випадків та травм. Крім того, кібербезпека є значущою проблемою для РТС, особливо тих, що підключені до Інтернету. Зі зростанням зв'язку цих систем вони стають вразливими до кіберзагроз, що можуть призвести до компрометації конфіденційної інформації, несанкціонованого доступу та навіть фізичної шкоди. Наприклад, хакер може отримати контроль над роботом та використовувати його для завдання шкоди майну чи пошкодження здоров'я людей. На сьогоднішній день існує декілька досліджень і публікацій, що детально вивчають функційну та кібербезпеку РТС.

Автори [1] під час власного практичного дослідження низки РТС різних виробників виявили критичні проблеми з кібербезпекою в декількох роботах. У зазначеній роботі змістовно описані загрози, які несе в собі скомпрометована РТС, а також виявлені проблеми кібербезпеки цих систем. Серед більш конкретних практичних робіт можна виділити дослідження [12] автори якого розглядають колаборативного робота Franka Emika Panda у якості об’єкта дослідження та змогли не тільки аналітично оцінити рівень захищеності цього кобота, а також виявили конкретні вразливості та визначили компоненти кобота, через які ці вразливості можуть бути експлуатовані. У дослідженні [13] в якості РТС-об’єкта було розглянуто іншого кобота – Universal Robots UR3. За допомогою використання симулятора РТС автори дослідили безпеку процесу оновлення прошивки кобота. Цей аналіз безпеки виявив чотири досі невідомі вразливості в процесі оновлення ПЗ, які можуть призвести до повної компрометації кобота. Крім того, автори [9] теоретично та експериментально дослідили виклики та наслідки безпеки сучасних РТС. Розглянули стандартну архітектуру РТС та проаналізували її з точки зору системної безпеки. Була побудована модель зловмисника, за допомогою якої автори показали як зловмисник може скомпрометувати контролер робота і отримати повний контроль над ним, що може призвести до змін у виробничому процесі. Також були досліджені потенційні наслідки таких атак та експериментально оцінена стійкість широко розповсюджених РТС до кібератак.

В свою чергу, у статті [10] автори узагальнили попередньо зазначені існуючі досліджень в галузі безпеки та кібербезпеки РТС, а також проаналізували основні проблеми і труднощі, що перешкоджають розвитку безпеки роботів, включаючи недостатню обізнаність про безпеку, відсутність тестових стендів, слабка захищеність прошивки/протоколів роботів, а також обмеженість обчислювальних ресурсів.

З огляду на представлену архітектуру РТС (рис.7.1) та результатів аналізу досліджень, сформовано перелік загроз функційної та кібербезпеки РТС.

1) Відсутність перевірки цілісності конфігураційних файлів. Промислові роботизовані системи працюють відповідно до програми завдань, що завантажується програмістом за допомогою FTP або API. Однак у більшості випадків перевірка безпеки для визначення того, чи є програма-завдання безпечною чи ні [13] відсутня, що дає можливість зловмисникам вводити шкідливі команди управління або завантажувати програми-завдання з логічними бомбами (код, який змушує робота зупинитися або зачепити інші пристрої) для маніпуляцій з роботом.

2) Слабка перевірка цілісності комунікаційних пакетів. Перевірка цілісності в пакетах зв'язку слабка (наприклад, перевірка кількості символів або обчислення контрольної суми), що дозволяє зловмисникам легко маніпулювати пакетами зв'язку у заводській мережі. [1,9]

3) Відкритий текст або слабке шифрування даних, що передаються. Через недостатню обізнаність виробників роботів у питаннях безпеки та обмеженість обчислювальних ресурсів роботів, більшість чутливих даних передається у вигляді відкритого тексту або у слабко зашифрованому вигляді, що дозволяє зловмисникам прослуховувати ці дані і викрадати конфіденційну інформацію робота. [10]

4) Слабке шифрування сховища облікових даних. Облікові дані (пари логін-пароль) зберігаються у файлі зі слабким шифруванням (наприклад, XOR), що дозволяє зловмисникам зламати файл і отримати облікові дані всіх дійсних користувачів.

5) Облікові дані за замовчуванням або жорстко закодовані. Деякі роботи мають облікові дані за замовчуванням або жорстко закодовані, які можна знайти в загальнодоступному посібнику користувача або в коді прошивки.

6) Відсутність конфігурації налаштувань привілеїв. Система контролю доступу може бути тимчасово відключена при завантаженні робота, а більшість систем контролю доступу мають некоректно налаштовані права доступу до різних функцій, що дозволяє зловмисникам з низькими привілеями маніпулювати критично важливими функціями роботів. [1,9]

7) Зловживання правами на читання/запис файлів. Файлові системи деяких промислових роботів є відкритими для виробничої мережі через такі сервіси, як FTP, що створює ризик того, що файли робота, які зберігають конфіденційні дані можуть бути прочитані/записані зловмисниками, які не мають прав доступу до них.

8) Відсутність перевірок безпеки. Деякі роботи оснащені загальнодоступними КДР, що дозволяє стороннім розробникам використовувати їх для своїх розробок. Однак ці роботи не перевіряють безпеку КДР, що дозволяє зловмисникам розробляти шкідливе програмне забезпечення для маніпулювання роботами. [1]

9) Відсутність перевірки цілісності прошивки. Образи прошивки контролера та навчального планшету повинні бути завантажені під час завантаження робота. Однак робот не перевіряє цілісність завантаженої

прошивки, що дозволяє зловмисникам маніпулювати прошивкою і, таким чином, роботом/навчальним планшетом. [9]

10) Відсутність фізичного захисту інтерфейсів. Якщо зловмисник отримає доступ до контролера, він може виконати шкідливе корисне навантаження на локально підключеному USB-пристрої. [12]

Функційна та кібербезпека є критично важливими характеристиками РТС і наразі досліджена доволі поверхнево. На нашу думку, розробники, виробники та операційники РТС дарма нехтують впровадженням комплексних рішень забезпечення функційної та кібербезпеки РТС, адже впроваджуючи надійні протоколи, можна користуватися перевагами технологічних досягнень, мінімізуючи при цьому потенційні безпекові ризики.

## 7.4. Методи оцінювання функційної та кібербезпеки РТС

Під час дослідження проаналізованорізні методи оцінювання функційної та кібербезпеки. Класифікація цих методів за типом методу наведена в Таблиці 7.1. У сфері безпеки аналітичний метод може передбачати використання математичних моделей або логічних схем для виявлення потенційних ризиків безпеки та розробки стратегій управління ризиками. Наприклад, аналітик може використовувати аналіз дерева атак для виявлення потенційних загроз у критично важливій для безпеки системі та розробки стратегій запобігання цим загрозам. Експериментальний метод може передбачати проведення контрольованих експериментів для оцінки ефективності заходів безпеки або перевірки працездатності критично важливих для безпеки систем. Наприклад, експеримент може включати тестування можливої експлуатації вразливостей системи для оцінки реакції системи на такі дії. Аналітично-експериментальний метод може синергетично поєднувати елементи обох підходів для розробки і тестування нових технологій безпеки або захисту або вдосконалення існуючих.

Таблиця 7.1 – Класифікація методів оцінювання за типом методу

| Тип методу | Назва методу |
|---|---|
| Аналітичний | Аналіз дерева атак (Attack Tree Analysis (ATA)) |
| | Оцінка ризиків та вразливостей (Risks & vulnerabilities assessment (R&VA)) |
| | Блок-схеми надійності, захищеності та безпеки (Reliability (Safety) Block Diagrams (R(S)BD)) |
| Експериментально-аналітичний | IMECA (Intrusion Modes and Effect Criticality Analysis) |
| | STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privileges) |
| Експериментальний | Тестування на проникнення (ТнП) |
| | Ін’єктування несправностей (вразливостей) та варіативностей (Faults and Variabilities Injection Testing (F&VIT)) |

7.4.1. Аналіз дерева атак (Attack Tree Analysis (ATA))

Аналіз дерева атак (ATA) забезпечує формальний, методичний спосіб опису безпеки систем, заснований на різних атаках. Він представляє атаки на систему у вигляді деревовидної структури, де мета є кореневим вузлом, а різні способи досягнення цієї мети – листовими вузлами [14]. Етапи ATA відображені на рисунку 7.2.

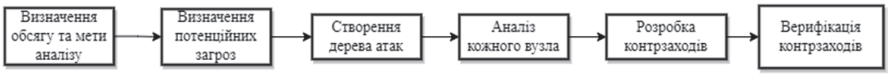



Рисунок 7.2 – Етапи аналізу дерева атак на РТС

1. Визначення обсягу та мети аналізу. Першим кроком є чітке визначення РТС, що аналізується та мети цього аналізу. Це допоможе зосередити аналіз на найбільш важливих загрозах.
2. Визначення потенційних загроз. Наступним кроком є мозковий штурм усіх потенційних загроз, які можуть виникнути в РТС. Це можна зробити за допомогою різних методів, таких як аналіз моделей атак.
3. Створення дерева атак. Дерево атак – ієрархічна діаграма, яка показує різні етапи атаки на РТС, від початкової точки входу до кінцевої мети зловмисника. Дерево будується шляхом розбиття сценарію атаки на менші, більш керовані частини.
4. Аналіз кожного вузла. Після створення дерева атак аналізується кожен вузол для визначення ймовірності та впливу атаки. Це допомагає визначити пріоритетність загроз і визначити, які з них потребують найбільшої та першочергової уваги.
5. Розробка контрзаходів. Після завершення аналізу розробляються контрзаходи для усунення виявлених загроз. Вони можуть варіюватися від технічних засобів контролю, таких як брандмауери та засоби контролю доступу, до процедурних засобів контролю, таких як підвищення обізнаності персоналу, що обслуговує та керує РТС, в питання функційної та кібербезпеки.
6. Верифікація контрзаходів. На останньому етапі ефективність контрзаходів тестується і перевіряється, щоб переконатися, що вони ефективно протидіють виявленим загрозам.

7.4.2. Оцінка ризиків та вразливостей (Risks & vulnerabilities assessment (R&VA))

R&VA-метод часто використовується для виявлення та оцінки потенційних слабких місць і вразливостей в РТС з метою розробки плану усунення та зменшення (пом’якшення) виявлених вразливостей і підтримання

постійної захищеності. Оцінка ризиків та вразливостей зазвичай складається з етапів зображених на рисунку 7.3.

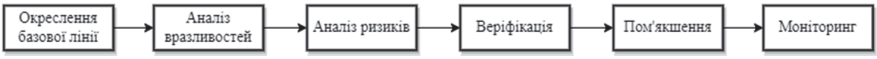



Рисунок 7.3 – Етапи оцінювання ризиків та вразливостей РТС

1. Окреслення базової лінії. Встановлення цілей керівництвом проведення оцінювання РТС перед його початком.
2. Оцінка вразливостей (ОВ). Проведення сканування вразливостей для виявлення слабких місць в РТС.
3. Оцінка ризиків (ОР). Класифікація вразливостей на основі рівня їх впливу та створення плану їх усунення.
4. Пом'якшення наслідків. Пом'якшення вразливостей відповідно до рівня їхнього впливу.
5. Верифікація. Перевірка ефективності плану щодо усунення вразливостей.
6. Моніторинг. Регулярний моніторинг та оновлення РТС для забезпечення постійної захищеності.

7.4.3. Блок-схеми надійності та безпеки (Reliability (Safety) Block Diagrams (R(S)BD))

Блок-схема надійності (RBD) – це метод графічного аналізу, який виражає відповідну систему у вигляді з'єднань ряду компонентів відповідно до їх логічного взаємозв'язку з точки зору надійності. Блок-схема безпеки (SBD) – це подібна техніка, що розглядає аспекти безпеки [16]. Побудова блок-схем надійності та безпеки є важливим інструментом для оцінки надійності та безпеки РТС і зазвичай складаються з етапів зображених на рисунку 7.4.

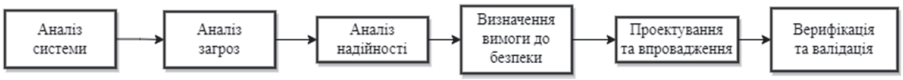



Рисунок 7.4 – Етапи побудови блок-схем надійності та безпеки РТС

Аналіз системи. Перший етап передбачає аналіз РТС для визначення її компонентів, підсистем та інтерфейсів. Це передбачає розуміння функцій системи, виявлення основних режимів збоїв і визначення наслідків цих збоїв.

1. Аналіз загроз. Другий етап передбачає визначення загроз, які можуть виникнути внаслідок збоїв у роботі РТС, включаючи будь-які потенційні загрози безпеці для людей або навколишнього середовища. Аналіз

небезпек зазвичай включає визначення потенційних загроз, оцінку ймовірності їх виникнення та оцінку тяжкості наслідків.

2. Аналіз надійності. Третій етап передбачає аналіз надійності компонентів і підсистем РТС. Це передбачає оцінку інтенсивності відмов і ймовірності відмови для кожного компонента, а також оцінку впливу стратегій резервування і відмовостійкого проектування.

3. Визначення вимог до безпеки. На четвертому етапі визначаються вимоги до безпеки РТС, включаючи будь-які критичні для безпеки функції, цільові показники надійності та заходи зі зниження ризиків. Ці вимоги можуть ґрунтуватися на галузевих стандартах, нормативних вимогах або очікуваннях зацікавлених сторін.

4. Проектування та впровадження. П'ятий етап передбачає проектування та впровадження РТС з урахуванням вимог до надійності та безпеки, визначених на попередніх етапах. Це може включати вибір відповідних компонентів, розробку стратегій резервування та відмовостійкості, а також тестування системи в різних умовах.

5. Верифікація та валідація. Заключний етап включає в себе перевірку та валідацію РТС, щоб переконатися, що вона відповідає вимогам надійності та безпеки. Це може включати тестування системи в нормальних і ненормальних умовах, проведення аналізу режимів відмов і перевірку відповідності відповідним стандартам і правилам.

Загалом, метою блок-схем надійності та безпеки є виявлення потенційних небезпек і режимів відмов, а також розробка систем, які мінімізують ризики, пов'язані з цими відмовами. Дотримуючись цих етапів, дизайнери та інженери можуть створювати більш надійні та безпечні РТС.

### 7.4.4. IMECA (Intrusion Modes and Effect Criticality Analysis)

Метою I(F)MECA-аналізу є оцінка такі характеристик системи, як доступність, безпека та вразливість РТС, використовуючи дві методики: IMECA (для вторгнень) та FMECA (для відмов) [17,18]. Основні етапи I(F)MECA відображені на рисунку 7.5.

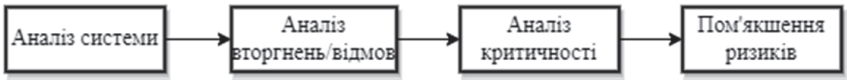



Рисунок 7.5 – Етапи I(F)MECA-аналізу компонентів РТС

1. Аналіз системи. На цьому етапі РТС розбивається на компоненти і кожен з них аналізується для виявлення потенційних режимів вторгнення/відмови.

2. Аналіз режимів вторгнення/відмов. Після того, як потенційні режими вторгнення/відмови визначені, вони аналізуються на предмет

визначення обсягу впливу на РТС. На цьому етапі розглядається, як кожен режим вторгнення/відмови може вплинути на продуктивність, надійність, безпеку, кібербезпеку або інші критичні аспекти системи.

3. Аналіз критичності. Після аналізу наслідків кожного режиму вторгнення/відмови, наступним етапом є визначення критичності кожного режиму вторгнення/відмови. Це передбачає присвоєння балів кожному типу вторгнення/збою на основі таких факторів, як ймовірність виникнення, серйозність впливу та можливість виявлення.

4. Пом'якшення ризиків. На основі оцінок критичності розробляються стратегії зниження ризиків, спрямовані на найбільш критичні режими вторгнень/збоїв. Це може включати перепроектування РТС, впровадження додаткових засобів захисту або резервування, або розробку планів на випадок надзвичайних ситуацій.

### 7.4.5. STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service and Elevation of privileges)

Методика STRIDE є відомим і широко використовуваним підходом до моделювання загроз, розробленим компанією Microsoft [19]. Ця абревіатура розшифровується як "підробка, втручання, відмова, розкриття інформації, відмова в обслуговуванні та підвищення привілеїв" (Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege). Кожен з цих термінів представляє потенційну загрозу або вектор атаки, який може бути використаний для компрометації безпеки РТС. Основні етапи методики STRIDE відображені на рис.7.6.

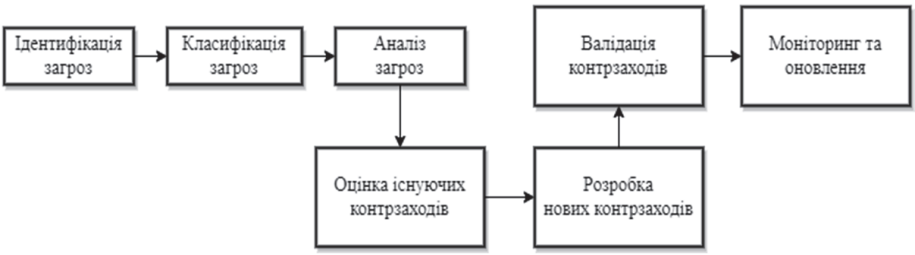



Рисунок 7.6 – Етапи STRIDE

1. Ідентифікація загроз. Першим етапом методики STRIDE є визначення загроз, які потенційно можуть поставити під загрозу безпеку РТС. Для кожної з категорій STRIDE розглядаються шляхи, через які система може бути атакована.

2. Класифікація загроз. Після того, як потенційні загрози визначені, їх необхідно класифікувати за ступенем серйозності та ймовірності. Це допоможе розставити пріоритети і зосередитися на найбільш критичних загрозах.

3. Аналіз загроз. Детальний аналізуй кожної з виявлених загроз. На цьому етапі оцінюється який вплив матиме загроза на систему в разі її реалізації, а також ймовірність її виникнення.

4. Оцінка існуючих контрзаходів. На цьому етапі визначаються та оцінюються ефективність існуючі контрзаходи для пом'якшення виявлених загроз.

5. Розробка нових контрзаходів. Якщо існуючих контрзаходів недостатньо, розробляються нові, щоб пом'якшити виявлені загрози. Ці контрзаходи можуть включати засоби контролю безпеки, процедури або політики.

6. Валідація контрзаходів. Після того, як були розроблені нові контрзаходи, вони мають бути впроваджені в РТС, а їх ефективність має бути додатково провалідована.

7. Моніторинг та оновлення. РТС потребує систематичного моніторингу на предмет виявлення нових загроз. Це допоможе гарантувати, що система залишатиметься безпечною протягом тривалого часу.

### 7.4.6. Тестування на проникнення (ТнП)

Тестування на проникнення (ТнП) широко використовуваний методологічним підхід, що дозволяє оцінити захищеність РТС шляхом імітації реальної атаки [20]. ТнП передбачає пошук комбінацій вразливостей в системі, які можуть бути використані для проведення атак на цільову систему. Якісно проведене тестування дозволяє визначити рівень захищеності системи та наявність в ній вразливостей, виявити найбільш ймовірні шляхи порушення встановленої політики безпеки, а також визначити, наскільки добре працює комплексність засобів захисту такої системи [21]. Використання ТнП також може бути корисним для визначення [22]:

1. Рівня захищеності РТС від можливих атак.

2. Ймовірний рівень досвідченості, необхідний зловмиснику для проведення успішних атак на РТС.

3. Здатності спеціалістів з кібербезпеки виявляти атаки та реагувати відповідним чином.

4. Додаткових контрзаходи, які можуть зменшити або виключити виявлені загрози РТС.

Стандарт NIST SP-800-115 описує декілька заходів тестування безпеки і надає настанови щодо планування та проведення різних видів заходів оцінки безпеки. Серед усього, зазначений стандарт визначає модель методології ТнП, яка базується на наступних чотирьох етапах, які повторюються ітеративно: планування, виявлення, атака, звітність (рисунок 7.7).

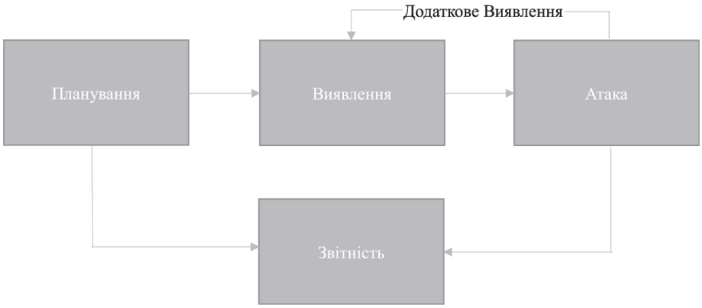



Рисунок 7.7 – Чотирьохетапна модель методології ТнП стандарту NIST SP-800-115

Цілі ТнП встановлюються на етапі планування. Етап виявлення охоплює збір інформації, сканування та аналіз вразливостей. Фаза атаки — це основний етап ТнП, під час якого перевіряються та підтверджуються раніше ідентифіковані вразливості. Фаза звітування відбувається одночасно з іншими трьома фазами ТнП. [23]

Penetration Testing Execution Standard (PTES) [24] це відкритий документ, який був розроблений, щоб забезпечити зв'язок між бізнесом і постачальниками послуг безпеки для виконання заходів ТнП. Він розділений на сім основних загальних розділів: (i) взаємодії перед залученням, (ii) збір інформації, (iii) моделювання загроз, (iv) аналіз вразливостей, (v) експлуатація, (vi) після експлуатація та (vii) звітування (рисунок 7.8):

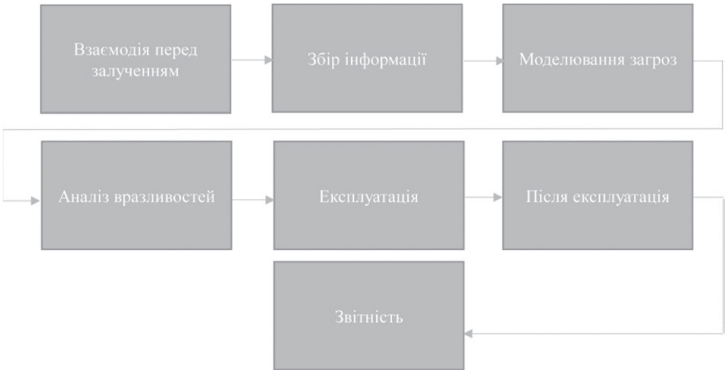



Рисунок 7.8 – Шестиетапна модель методології ТнП зі стандарту PTES

The Open-Source Security Testing Methodology Manual (OSSTMM) [25] це ще одна відкрита методологія оцінки безпеки, запропонована Institute for Security and Open Methodologies. Ця методологія дозволяє проводити аудит

взаємодії людей, систем і мережевих комунікацій з метою оцінки векторів атаки та надійності заходів безпеки. OSSTMM не надає переліку інструментів для фактичного тестування кожної області, але визначає, що потрібно перевірити і що робити перед, під час і після перевірки безпеки, а також виступає в якості допоміжного посилання в кількох процесах сертифікації безпеки. Робочий процес OSSTMM розділений на чотири етапи: (i) Введення, (ii) Взаємодія, (iii) Розслідування та (iv) Втручання (рисунок 7.9):

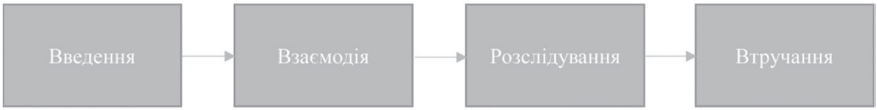



Рисунок 7.9 – Чотирьохетапна модель методології ТнП стандарту OSSTMM

Типи випробувань і властивості безпеки, які мають підлягати перевірці, визначаються на етапі введення. На етапі взаємодії тестувальник визначає та вибирає цільові системи. Під час фази розслідування тестувальник забирає якомога більше інформації про цільові активи. Лише на останній фазі він займається безпосередньо перевіркою безпеки системи. [23]

The Information Systems Security Assessment Framework (ISSAF) [26] це ще одна методологія, розроблена для оцінки засобів контролю безпеки мережі, системи та програм. Його структура визначає три основні етапи: (i) Планування та підготовка, (ii) Оцінка, (iii) Звітування, очищення та знищення артефактів (рисунок 7.10).

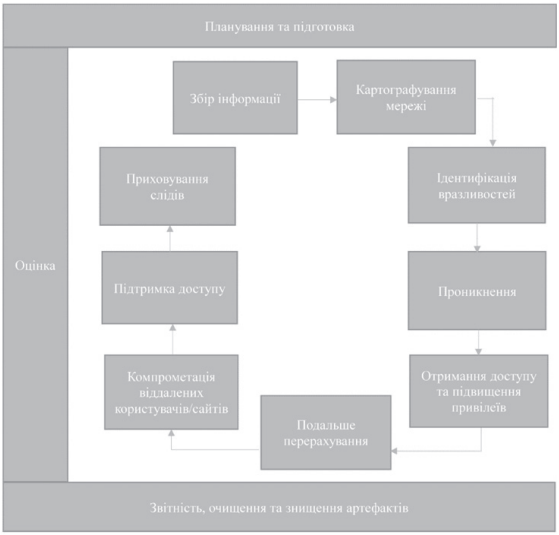



Рисунок 7.10 – Модель методології ТнП стандарту ISSAF

На першому етапі встановлюються цілі для оцінки безпеки та плануються тести, які мають бути проведені під час другого етапу. У свою чергу, другу фазу можна далі поділити на наступні операційні підфази: (i) збір інформації, (ii) картографування мережі, (iii) ідентифікація вразливостей, (iv) проникнення, (v) отримання доступу та підвищення привілеїв, подальше перерахування (vi), компрометація віддалених користувачів/сайтів (vii), підтримка доступу (viii), приховування слідів (ix). Третій етап охоплює процес звітування та видаляє будь-які артефакти, що залишилися від фактичного етапу ТнП.

Проаналізувавши кожен з розглянутих підходів до ТнП можемо виділити наступні ключові 5 етапів ТнП, зображені на рисунку 7.11.


Рисунок 7.11 – Ключові етапи ТнП РТС

1. Планування та розвідка. Цей етап передбачає збір інформації про цільову систему, наприклад, про топологію мережі, IP-адреси та додатки, для виявлення потенційних точок входу та вразливостей.
2. Сканування. На цьому етапі використовуються різні інструменти сканування для виявлення та складання карти вразливостей цільової системи, таких як відкриті порти, сервіси та потенційні вразливості в додатках або операційній системі.
3. Отримання доступу до РТС. Після виявлення потенційних вразливостей робляться спроби використати їх для отримання доступу до системи або додатку.
4. Підтримка доступу до РТС. Якщо доступ успішно отримано, тестер проникнення намагається утримувати доступ до системи якомога довше, щоб надалі оцінити її безпеку.
5. Аналіз результатів і звітність. Результати ТнП аналізуються та створюється звіт в якому детально описуються виявлені вразливості, методи, використані для їх виявлення, а також рекомендації щодо пом’якшення та/або усунення цих вразливостей.

### 7.4.7. Тестування ін’єктування відмов та варіативностей (Faults and Variabilities Injection Testing (FVI-тестування))

FVI-тестування є загальновживаною експериментальною методикою для оцінки надійності мікропроцесорних систем [27], таких як РТС. Загальні етапи FVI-тестування зображені ра рисунку 7.12.

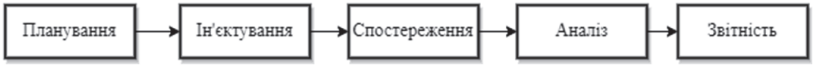



Рисунок 7.12 – Етапи FVI-тестування РТС

1. Планування. На цьому етапі визначається стратегія FVI-тестування. Це включає визначення компонентів РТС, що підлягають тестуванню, вибір типів несправностей та варіативних змін, які необхідно ввести, а також визначення середовища та умов проведення тестування.

2. Ін’єктування. На цьому етапі в систему вносяться помилки і зміни. Типи несправностей і змін, які можуть бути введені, включають апаратні несправності (наприклад, помилки пам'яті і збої диска), програмні несправності (наприклад, помилки кодування і логічні помилки), а також зміни навколишнього середовища (наприклад, затримки в мережі і коливання потужності).

3. Спостереження. На цьому етапі спостерігається реакція системи на введені несправностей та змін. Це передбачає моніторинг поведінки системи, збір показників продуктивності та виявлення будь-яких неочікуваних дій або помилок.

4. Аналіз. На цьому етапі дані, зібрані під час спостереження, аналізуються для виявлення закономірностей і першопричин збоїв і відхилень. Це допомагає визначити області РТС, які потребують вдосконалення, і знайти потенційні рішення.

5. Звітність. Нарешті, створюється звіт, який підсумовує результати FVI- тестування. Звіт може містити рекомендації щодо підвищення стійкості та надійності системи, а також будь-які виявлені області для подальшого тестування та аналізу.

## 7.5. Комбінування методів оцінювання безпеки та кібербезпеки РТС

### 7.5.1. Принципи і модель комбінування

Під час нашого дослідження ми проаналізували кожен етап усіх розглянутих методів оцінки функційної та кібербезпеки РТС і розробили модель комбінування методів, яка відображена на рисунку 7.13.

Ми узагальнили схожі етапи та згрупували їх, наприклад, ми помітили, що всі 7 методів мають початковий етап, коли фахівці виконують завдання/діяльність з попередньої обробки без прямої взаємодії з цільовою системою, наприклад, планування, ідентифікація компонентів тощо. Більшість інших етапів оброблялися так само, але ми також виділили такий тип можливих етапів тестування, як «Підготовка системи», що описує додаткові кроки, які необхідно виконати (наприклад, ін'єктування) перед початком процесу оцінювання.

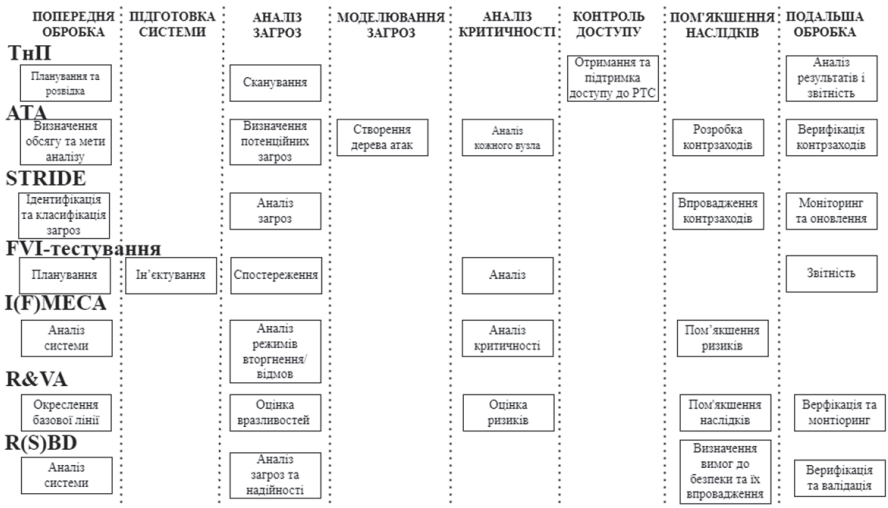



Рисунок 7.13 – Методи оцінювання функційної та кібербезпеки

### 7.5.2. Аналіз варіантів комбінування методів

Вибір методу залежить від конкретної проблеми, що вирішується, та наявних ресурсів. Використання комбінації підходів може допомогти забезпечити ефективну ідентифікацію та управління ризиками безпеки, а також надійність та ефективність критично важливих для безпеки РТС. Враховуючи той фактор, що загальна кількість комбінацій дорівнює 21 (при умові поєднання тільки двох методів) та більше, у цьому розділі проаналізовано та оцінено лише декілька найрозповсюдженіших варіантів поєднання існуючих методів оцінки безпеки та кібербезпеки РТС.

#### 7.5.2.1. I(F)MECA-ТнП

Така комбінація методів повністю застосовна для такої критично важливих систем, як РТС. Всі можливі поверхні атаки (позначені пунктирним овалом на рис.1), такі як мережі роботів/сервісів, контролери або програмне завдання, можуть бути оцінені за допомогою цієї комбінації методів. I(F)MECA-ТнП дозволяє фахівцям оцінювати і виконувати аналіз вторгнень/відмов, а також оцінювати критичність результатів (загроз/вразливостей/атак) після фаз ТнП, таких як сканування, отримання доступу і підтримка доступу. На рисунку 7.14 показано, як етапи I(F)MECA можуть бути інтегровані в потік ТнП.


Рисунок 7.14 – Інтеграція I(F)MECA-аналізу в етапи ТнП

7.5.2.2. ATA-ТнП

ATA дуже часто використовується як доповнення до ТнП, особливо після етапу сканування, що дозволяє точніше виявити потенційні загрози та схематично відобразити їх та проаналізувати з метою полегшення етапу отримання доступу до РТС. Етапи ATA-ТнП описані на рисунку 7.15.


Рисунок 7.15 – Інтеграція ATA в етапи ТнП

7.5.2.3. R&VA-ТнП

Це поєднання методів настільки поширене, що іноді їх плутають між собою. R&VA-ТнП - це дуже масштабний і довготривалий процес, який охоплює не лише оцінку поточного стану функційної та кібербезпеки, але й після-моніторинг вразливостей/ризиків на предмет появи нових. Етапи R&VA-ТнП описані на рис. 7.16.

7.5.2.4. FVI-ТнП

Безперечно, унікальним етапом методу FVI-ТнП є етап ін'єктування. Цей метод може бути корисним для перевірки повноти та надійності процесу ТнП. Ідея полягає в тому, що команда розробників вводить певні відомі вразливості/помилки/варіативні зміни перед початком етапу тестування, а

фахівці, що проводять тестування РТС мають перевірити, як система відреагує на їх появу і до яких наслідків вони можуть призвести. На жаль, цей метод є досить ризикованим, оскільки ін'єктування може призвести до неочікуваної поведінки РТС, тому не рекомендується проводити його на реальній існуючій системі. Етапи FVI-ТнП детально обґрунтовані на рисунку 7.17. Отримання та підтримка доступу може бути необов'язковим етапом.

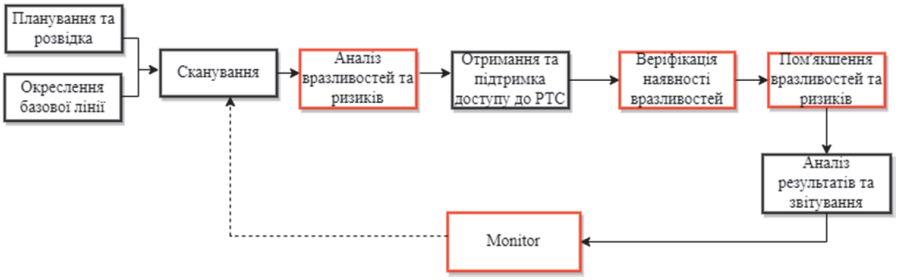



Рисунок 7.16 – Інтеграція R&VA в етапи ТнП

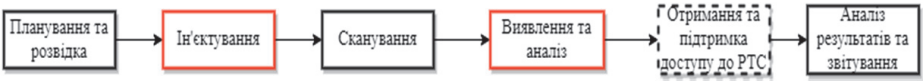



Рисунок 7.17 – Інтеграція FVI в етапи ТнП

7.5.2.5. Аналіз недоліків та переваг оглянутих поєднань методів

Ми порівняли та обговорили комбінації методів і підсумували їхні переваги та недоліки в таблиці 7.3.

Таблиця 7.3 – Переваги та недоліки комбінацій методів оцінювання функційної та кібербезпеки РТС

| Переваги | Недоліки |
|---|---|
| I(F)MECA-ТнП | |
| Ця комбінація методів допомагає спрогнозувати вплив виявлених вразливостей, не валідуючи ці знахідки. У випадках, коли метою є оцінка впливу можливих загроз, цей метод може заощадити багато часу шляхом виключення сценаріїв низької критичності зі сфери тестування. | Деякі виявлені вразливості можуть бути "відомими" проблемами, що вже мають відповідні ідентифікатори CVE в базах вразливостей, таких як NVD/MITRE та рівень їх критичності вже оцінений, а кроки експлуатації детально описані. |

Продовження табл. 7.3

| | |
|---|---|
| Отримавши доступ до РТС, фахівці можуть повторити аналіз I(F)MECA, щоб виявити найбільш критичні загрози і почати процес експлуатації з них. | Якщо доступ до РТС отримано на дуже короткий період, фахівець з тестування може не встигнути провести аналіз I(F)MECA. |
| Ця комбінація методів також стане в нагоді після регресійного тестування, щоб зрозуміти рівень пом'якшення та/або усунення ризиків/вразливостей. | - |
| ATA-ТнП | |
| Ця комбінація методів допомагає фахівцям визначити можливі вектори атаки на РТС, такі як мережі кобота/сервісів, контролер або програмне завдання, і проаналізувати їх компоненти та виявити найефективнішій шлях вторгнення в РТС. | Потребує додаткового часу для створення та аналіз дерева атак. |
| Застосовані контрзаходи можуть бути перевірені за допомогою цієї комбінації методів. | - |
| R&VA-ТнП | |
| Ця комбінація методів найширше покриття тестування серед оглянутих. | Витратна та трудомістка комбінація методів. |
| Спрощує пом'якшення наслідків «хибно позитивних» результатів, перевіряючи їх за допомогою експлуатаційних заходів. | - |
| FVI-ТнП | |
| Використовуючи цю комбінацію методів, команда розробників може побачити, як РТС буде працювати в несподіваних/критичних ситуаціях. | Ризикована комбінація методів оцінювання, якщо використовувати на компонентах реально-існуючої РТС. |
| Поєднання цих методів дозволяє оцінити якість роботи фахівці з тестування РТС. | Трудомістка комбінація методів, бо потребує додаткового часу на розробку шляхів ін'єктування помилок/вразливостей та їх безпосереднє впровадження в РТС. |

## 7.6. Приклад вибору варіанту

Повнота, достовірність, час виконання і вартість є важливими показниками безпеки РТС. Повнота - це показник, що визначає, чи всі можливі ризики безпеки були виявлені і розглянуті в процесі оцінювання. Іншими словами, оцінка повинна виявити всі потенційні загрози/вразливості і допомагати запропонувати ефективні заходи для їх усунення. Вкрай важливо досягти високого ступеня повноти оцінювання, оскільки навіть один пропущений ризик може призвести до серйозних наслідків.

Достовірність свідчить про правдивість і точність результатів оцінювання. Оцінка повинна ґрунтуватися на надійній методології та проводитися кваліфікованими фахівцями. Процес оцінювання також має бути прозорим і добре задокументованим, щоб полегшити незалежну перевірку результатів. Достовірність важлива, оскільки зацікавлені сторони повинні довіряти результатам оцінки, щоб приймати обґрунтовані рішення про використання роботизованих систем.

Час виконання - це тривалість процесу оцінювання. Оцінювання повинно проводитися ефективно без шкоди для якості та повноти результатів. Тривалий процес оцінювання може затримати розгортання РТС і збільшити витрати. Тому час виконання має важливе значення для забезпечення своєчасного і економічно ефективного проведення оцінювання.

Вартість - це фінансові ресурси, необхідні для проведення оцінювання. Вартість повинна бути обґрунтованою і пропорційною вартості РТС, що оцінюється. Хоча вартість є важливим фактором, вона не повинна ставити під загрозу якість, повноту і достовірність оцінювання.

У випадку спільного оцінювання безпеки роботів і кібербезпеки пріоритетність цих параметрів залежатиме від конкретного контексту і вимог оцінювання. Варіант рейтингу пріоритетів відображено в таблиці 7.4 у вигляді аналітичних вагових коефіцієнтів.

Таблиця 7.4 – Рейтинг пріоритетів показників оцінювання

| Приклад РТС\Показники | Повнота | Достовірність | Час виконання | Вартість | Сума |
|---|---|---|---|---|---|
| Кобот | 0.35 | 0.3 | 0.2 | 0.15 | 1.0 |

Також, на основі аналізу можливих варіантів комбінування існуючих методів забезпечення функційної та кібербезпеки РТС та огляду переваг і недоліків цих методів можна побудувати узагальнену матрицю впливу на показники оцінювання. У запропонованому прикладі вплив комбінації методів на показники оцінювання функційної та кібербезпеки РТС визначено за шкалою від -2 до 2, де -2 – суттєве погіршення метрики, -1 – несуттєве погіршення метрики, 0 – значення метрики не змінюється, 1 – несуттєве збільшення метрики, 2 – сутєєве збільшення метрики. Результати незваженого та зваженого оцінювання надано в таблиці 7.5.

Таблиця 7.5 – Оцінка комбінація методів оцінювання функційної та кібербезпеки РТС

| Комбінація\ Вплив | Повнота | Достовірність | Час виконання | Вартість | Загальна оцінка | |
|---|---|---|---|---|---|---|
| | | | | | незважена | зважена |
| FVI-ТнП | 1 | 2 | -1 | -1 | 1 | 0,60 |
| I(F)MECA-ТнП | 2 | 2 | -1 | -1 | 2 | 0,95 |
| ATA-ТнП | 1 | 2 | -1 | -1 | 1 | 0,60 |
| R&VA-ТнП | 2 | 1 | -1 | -2 | 0 | 0,50 |

I(F)MECA-ТнП не суттєво збільшує час виконання та вартість оцінювання, але незрівнянно покращує його повноту та достовірність. У той же час, комбінація ATA-ТнП також збільшує час і вартість проведення оцінювання, але також дещо покращує його повноту і достовірність. R&VA-ТнП – це комбінація методів, що вимагає значних витрат часу та коштів, але значно покращує повноту оцінювання. FVI-ТнП – це комбінація методів, яка вимагає використання додаткового середовища для проведення тестування, але цей метод може значно підвищити надійність РТС.

## 7.7. Висновки

Під час дослідження було проаналізовано архітектуру РТС, її компоненти та особливості, а також потенційні загрози функційної та кібербезпеки РТС.

З метою аналізу та пошуку варіантів комбінування методів оцінювання функційної та кібербезпеки РТС були розглянуті 7 методів та запропоновані 4 комбінації цих методів: I(F)MECA-ТнП, ATA-ТнП, R&VA-ТнП та ATA-ТнП. Попередньо визначено, що поєднання I(F)MECA з ТнП, запропоноване в [28], є найкращим з розглянутих, що потребує практичного підтвердження в умовах застосування в реальній РТС або її емуляторі. До напрямів подальших досліджень доцільно віднести:

1) розширення переліку комбінацій методів розділу 7.5, формалізації їхнього операційного об’єднання та оцінювання;

2) вдосконалення методики оцінювання впливу комбінацій методів на такі показники оцінювання функційної та кібербезпеки РТС, як повнота оцінювання та його достовірність, час виконання, та вартість;

3) аналіз стимуляторів РТС щодо можливості повноцінного експериментування задля дослідження безпекових завдань, а також пошук симулятора РТС, який має бути використаний для підтвердження або спростування попередніх аналітичних оцінок розділу 7.6.1 та ін.

Література

1. Cerrudo C., Apa L. Hacking Robots before Skynet, 2017. Режим доступу: https://ioactive.com/pdfs/Hacking-Robots-Before-Skynet.pdf (станом на травень 2023 року).

2. TheVerge.com. Mall security bot knocks down toddler, breaks Asimov's first law of robotics. http://www.theverge.com/2016/7/13/12170640/mall-security-robot-k5-knocks-down-toddler (станом на травень 2023 року).

3. Mashable.com. Cute Chinese robot loses control, smashes window and injures someone. Режим доступу: http://mashable.com/2016/11/21/xiao-pang-chinese-robot-smashes-glass/#nrErRt0Pe5qX (станом на травень 2023 року).

4. Wired.com. Robot Cannon Kills 9, Wounds 14. Режим доступу: https://www.wired.com/2007/10/robot-cannon-ki/ (станом на травень 2023 року).

5. BBC.com. Robotic surgery linked to 144 deaths in the US. Режим доступу: http://www.bbc.com/news/technology-33609495 (станом на травень 2023 року).

6. Zhang P. Industrial control engineering // Advanced Industrial Control Technology, 2010, Ch. 2, pp. 41–70. Режим доступу: https://doi.org/10.1016/B978-1-4377-7807-6.10002-6 (станом на травень 2023 року).

7. Hägele M., Nilsson K., Norberto Pires J. Industrial Robotics // Springer Handbook of Robotics, 2007, pp. 963–986. Режим доступу: 10.1007/978-3-540-30301-5_43 (станом на травень 2023 року).

8. IFR.org. IFR presents World Robotics 2021 reports. Режим доступу: https://ifr.org/ifr-press-releases/news/robot-sales-rise-again (станом на травень 2023 року).

9. Quarta D., Pogliani M., Polino M., Maggi F., Zanchettin A. M., Zanero S. An Experimental Security Analysis of an Industrial Robot Controller // 2017 IEEE Symposium on Security and Privacy (SP), 2017, pp. 268–286. Режим доступу: 10.1109/SP.2017.20 (станом на травень 2023 року).

10. Pu H., He L., Cheng P., Sun M., Chen J. Security of Industrial Robots: Vulnerabilities, Attacks, and Mitigations // IEEE Network, 2022, pp. 1–12, Режим доступу: 10.1109/MNET.116.2200034 (станом на травень 2023 року).

11. Chan CF., Chow KP., Tang T. Security Analysis of Software Updates for Industrial Robots // Critical Infrastructure Protection XV. ICCIP 2021. IFIP Advances in Information and Communication Technology, Staggs J. and Shenoi J., Ed. Jan. 2022, vol. 636, pp. 229–245. Режим доступу: 10.1007/978-3-030-93511-5_11 (станом на травень 2023 року).

12. Hollerer S., Fischer C., Brenner B., Papa M., Schlund S., Kastner W., Fabini J., Zseby T. Cobot attack: a security assessment exemplified by a specific collaborative robot // Procedia Manufacturing, 2021, Vol. 54, pp. 191–196. Режим доступу: 10.1016/j.promfg.2021.07.029 (станом на травень 2023 року).

13. Pogliani M., Maggi F., Balduzzi M., Quarta D., Zanero S. Detecting Insecure Code Patterns in Industrial Robot Programs // 15th ACM Asia Conference

on Computer and Communications Security, pp. 759–771, Oct. 2020. Режим доступу: 10.1145/3320269.3384735 (станом на травень 2023 року).

14. Rahman A. Attack Trees: Cyber Security, 2020. Режим доступу: https://www.academia.edu/62051416/Attack_Trees_Cyber_Security (станом на травень 2023 року).

15. ECCouncil.org. Vulnerability Assessment: 6 Best Steps to Better Security. Режим доступу: https://egs.eccouncil.org/wp-content/uploads/2020/12/Risk-and-Vulnerability-Assessment-Do-You-Know-the-Other-Side.pdf (станом на травень 2023 року).

16. Sklyar V., Andrashov A., Babeshko E., Kovalenko A. Field Programmable Gate Array Technology for NPP I&Cs // Nuclear Power Plant Instrumentation and Control Systems for Safety and Security, 2014, Ch. 4, pp. 116–145. Режим доступу: 10.4018/978-1-4666-5133-3.ch004 (станом на травень 2023 року).

17. Torianyk V., Kharchenko V., Zemlianko H. IMECA Based Assessment of Internet of Drones Systems Cyber Security Considering Radio Frequency Vulnerabilities // IntelITSIS'2021: 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security, 2021. Режим доступу: https://ceur-ws.org/Vol-2853/paper50.pdf (станом на травень 2023 року).

18. Pevnev V., Torianyk V., Kharchenko V. Cyber Security of Wireless Smart Systems: Channels of Intrusions and Radio Frequency Vulnerabilities, Radioelectronic and Computer Systems 4, 2020, vol. 96, pp. 79–92. Режим доступу: doi:10.32620/reks.2020.4.07 (станом на травень 2023 року).

19. Mauri L., Damiani E. STRIDE-AI: An Approach to Identifying Vulnerabilities of Machine Learning Assets // 2021 IEEE International Conference on Cyber Security and Resilience, 2021. Режим доступу: https://doi.org/10.1109/CSR51186.2021.9527917 (станом на травень 2023 року).

20. Denis M., Zena C., Hayajneh T. Penetration testing: Concepts, attack methods, and defense strategies // 2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2016. Режим доступу: 10.1109/LISAT.2016.7494156 (станом на травень 2023 року).

21. Abakumov A., Kharchenko V. Penetration testing for IoT systems: cyber threats, methods, and stages // Electronic Modeling, 2022, vol. 44(4), pp. 79–104. Режим доступу: 10.15407/emodel.44.04.079 (станом на травень 2023 року).

22. Scarfone K., Souppaya M., Cody A. Orebaugh A. Technical Guide to Information Security Testing and Assessment: Recommendations of the National Institute of Standards and Technology // National Institute of Standards and Technology, Gaithersburg, MD, NIST Special Publication 800-115. Режим доступу: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf (станом на травень 2023 року).

23. Rak M., Salzillo G., Romeo C. Systematic IoT Penetration Testing: Alexa Case Study // ITASEC, 2020, vol. 2597(17). Режим доступу: http://ceur-ws.org/Vol-2597/paper-17.pdf (станом на травень 2023 року).

24. PTES Technical Guidelines: The Penetration Testing Execution Standard, 2017. Режим доступу: http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines (станом на травень 2023 року).

25. Herzog P. OSSTMM 3: The open-source security testing methodology manual-contemporary security testing and analysis, 2010. Режим доступу: https://www.isecom.org/OSSTMM.3.pdf (станом на травень 2023 року).

26. Busleiman A., Martorella C., Sarrazyn D., Racciatti H.M., Asgarally K. Information Systems Security Assessment Framework (ISSAF), 2015. Режим доступу: https://untrustednetwork.net/files/issaf0.2.1.pdf (станом на травень 2023 року).

27. Aponte-Moreno A., Isaza-GonzÃlez J., Serrano-Cases A., Martínez-Álvarez A., Cuenca-Asensi S., Restrepo-Calle F. Evaluation of fault injection tools for reliability estimation of microprocessor-based embedded systems // Microprocessors and Microsystems, 2023, vol. 96, pp. 1-13. Режим доступу: 10.1016/j.micpro.2022.104723 (станом на травень 2023 року).

28. Abakumov A., Kharchenko V. Combining IMECA analysis and penetration testing to assess the cybersecurity of industrial robotic systems // 12th IEEE Conference on Dependable Systems, Services and Technologies, DESSERT, 2022. Режим доступу: 10.1109/DESSERT58054.2022.10018823 (станом на травень 2023 року).

# 8. МЕТОДИ ЗАБЕЗПЕЧЕННЯ КІБЕРББЕЗПЕКИ ВЕБ-СИСТЕМ З ВИКОРИСТАННЯМ ЗАСОБІВ ШТУЧНОГО ІНТЕЛЕКТУ

**Д. О. Сверчков, Г. В.Фесенко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 8.1. Вступ

Високі темпи впровадження Інтернету у життя підприємств, установ та організацій сприяють активному використанню ними веб-застосунків і веб-сервісів. Веб-застосунки дозволяють організаціям збільшити свої доходи та покращити свої бізнес-процеси (наприклад, за рахунок віртуалізації або бізнес-платформи в ланцюжку поставок), а веб-сервіси використовуються для надання послуг у різних галузях – від онлайн калькуляторів до хмарних сховищ.

Доступність і надмірне використання роблять веб-застосунки мішенню для кібератак, які можуть призвести, зокрема, до:

- недоступності веб-застосунку внаслідок надсилання на нього зловмисного запиту;
- компрометації вразливих веб-застосунків з метою поставити під загрозу конфіденційність, цілісність і доступність ресурсів організації;
- пошкодження веб-застосунку шляхом знищення певних ресурсів, викрадення важливої інформації з баз даних, організації збоїв в роботі служби, створення проблем з доступом до веб-застосунків.

Серед основних ризиків безпеки веб-застосунків автори [1] відзначають ін'єкцію мови структурованих запитів (Structured Query Language (SQL)), міжсайтовий скриптинг (Cross Site Scripting (XSS)), включення на стороні сервера, а також порушену автентифікацію.

Успішній реалізації кібератак можуть сприяти записи зловмисниками натискань клавіш, крадіжки файлів cookie браузера, завантаження з мережі ботнет недосвідченими або неуважними користувачами зловмисного програмного забезпечення (ПЗ).

Застосовувані зараз для забезпечення безпеки веб-застосунків та веб-сервісів антивіруси і системи виявлення вторгнень (Intrusion Detection System (IDS)) використовують підходи на основі підпису та евристики. Евристичні підходи, такі як аналіз файлів, емуляція файлів і загальне виявлення сигнатур, базуються на наборі експертних правил прийняття рішень. Однак ці підходи мають низку недоліків. По-перше, знання про нову атаку є важливим для оновлення застосунку для ефективного виявлення, що створює вікно вразливості, яке зловмисники можуть використовувати для запуску атаки нульового дня та інші типи атак. По-друге, тривалий час аналізу та сканування

у поєднанні з великою кількістю помилкових спрацьовувань значно знижують продуктивність.

Для підвищення ефективності протидії кіберзагрозам все частіше залучаються сучасні технології, зокрема, особливо багато уваги приділяється методам штучного інтелекту (ШІ). Застосування ШІ надає можливість виявляти кіберзагрози в режимі реального часу на основі аналізу даних, що надходять з різних джерел. Алгоритми ШІ постійно адаптуються та оновлюються, щоб виявляти загрози ще до того, як зловмисники зможуть виявити вразливість у захисті мережі організації та реалізувати свої наміри. Іншими словами, алгоритми ШІ «розуміють» усі нюанси інфраструктури та мережі організації, а також дозволяють прогнозувати можливі сценарії атак. Отже, попит на рішення, засновані на використанні ШІ для виявлення кіберзагроз, постійно зростає.

Метою досліджень є ґрунтовний аналіз літературних джерел, присвячених питанням застосування ШІ у кібербезпеці, та формулювання рекомендацій щодо його застосування для забезпечення кібербезпеки веб-сервісів.

## 8.2. Класифікація джерел

Провівши аналіз відібраних літературних джерел [1-31], дійшли висновку, що їх можна розподілити за наступними типами:

– оглядові статті, тобто статті, які роблять загальний огляд існуючих джерел із зазначеної тематики, іноді фокусуючись на конкретних рішеннях та прикладах їх застосування [1-10];

– джерела, що пропонують рішення «все в одному» [14]. Дуже рідкісний вид публікацій, представлений великими за обсягом статтями, у яких автори розглядають дуже широкий спектр інформації: від основ кібербезпеки і використання у ній штучного інтелекту до конкретних методів і моделей із прикладами і рекомендаціями з використання, а також пропонують результати досліджень зі швидкодії і якості реагування на кібератаки;

– статті з описанням методів і моделей. Такий тип статей є найпопулярнішим. Статті такого типу статті описують розроблені моделі і методи та надають рекомендації щодо їх використання та впровадження [11-26].

Саме на основі аналізу останнього типу статей було проведено подальшу класифікацію джерел за способом використання ШІ у кібербезпеці та виокремлено два наступних типи джерел:

 джерела, що описують використання застосунків на основі ШІ для аналізу і оцінки існуючих систем на вразливості [11-15]. У таких джерелах, наприклад, можуть розглядатися особливості застосування спеціального ПЗ для

проведення статичного аналізу вихідного коду, аналізу системи з імітацією атаки, класифікації систем з метою їх подальшого аналізу в специфічних аналізаторах, а також оцінювання ступеня захищеності системи;

 джерела, що описують використання вбудованих механізмів ШІ для пошуку, виявлення, класифікації і боротьби з атаками на систему під час її роботи [16-26]. Такі джерела, наприклад, можуть розглядати питання розроблення та застосування низькорівневих моделей і методів виявлення: підозрілих даних на рівні запитів та протоколів; шкідливого коду; підозрілих дій та поведінки користувачів.

## 8.3. Аналіз джерел за напрямами досліджень

Далі представимо стислу характеристику змісту взятих до розгляду джерел, які віднесені до: оглядових статей (таблиця 8.1); джерел, що описують використання застосунку на основі ШІ для аналізу і оцінки існуючих систем на вразливості (таблиця 8.2); джерел, що описують використання вбудованих механізмів ШІ для пошуку, виявлення, класифікації і боротьби з атаками на систему під час її роботи (таблиця 8.3).

Таблиця 8.1 – Стисла характеристика змісту взятих до розгляду оглядових статей

| 1. Джерело | 2. Рік | 3. Що запропоновано | 4. Коментарі |
|---|---|---|---|
| 5. A Role of Artificial Intelligence Techniques in Networking [1] | 6. 2012 | 7. Огляд ролі ШІ в безпеці мережі | 8. |
| 9. Artificial intelligence for cybersecurity a systematic mapping of literature [2] | 10. 2020 | 11. Загальний огляд ШІ у кібербезпеці | 12. |
| 13. Artificial Intelligence in Cyber Security [3] | 14. 2021 | 15. Огляд реалізацій ШІ у різних системах кібербезпеки та оцінка перспективи підвищення ефективності кіберзахисту шляхом використання ШІ | 16. |

Продовження табл. 8.1

| | | | |
|---|---|---|---|
| 17. Detecting Cybersecurity Attacks in Internet of Things Using Artificial Intelligence Methods A Systematic Literature Review [4] | 18. 2022 | 19. Огляд методів глибокого навчання (Deep Learning (DL)) і машинного навчання (Machine Learning (ML)), які використовуються для забезпечення кібербезпеки Інтернету речей з поданням результатів оцінювання їх ефективності для виявленні атак. Огляд IDS з інтелектуальними архітектурними структурами, які використовують ШІ | 20. Виявлено, що опорні векторні машини (Support Vector Machine (SVM)) і випадковий ліс (Random Forest (RF)) є одними з найбільш використовуваних методів через високу точність виявлення |
| Detecting cyber threats through social network analysis: short survey [5] | 2017 | Огляд кіберзагроз у соціальних мережах | |
| Difficulties Faced and Applications of Machine Learning in Cyber-Security [6] | 2021 | Огляд прикладів застосування ML для протидії кіберзлочинцям | |
| Harnessing Artificial Intelligence Capabilities to Improve Cybersecurity [7] | 2020 | Огляд можливостей застосування ШІ для визначення сильних і слабких сторін запропонованих рішень з кібербезпеки | |
| Understanding the Strategic and Technical Significance of Technology for Security [8] | 2019 | Огляд технологій ШІ, які використовуються для забезпечення кібербезпеки | |

Продовження табл. 8.1

| | | | |
|---|---|---|---|
| Machine Learning and Cybersecurity [9] | 2020 | Огляд можливостей ML для забезпечення кібербезпеки | |
| Performance evaluation of Convolutional Neural Network for web security [10] | 2021 | Огляд можливостей DL для забезпечення безпеки веб-застосунків | |

Таблиця 8.2 – Стисла характеристика взятих до розгляду джерел, що описують використання застосунку на основі ШІ для аналізу і оцінки існуючих систем на вразливості

| 21. Назва | 22. Рік | 23. Що запропоновано | 24. Коментарі |
|---|---|---|---|
| 25. A Web Platform for Integrated Vulnerability Assessment and Cyber Risk Management [11] | 26. 2019 | 27. Онлайн-інструмент менеджменту і оцінки вразливостей ПЗ на основі стандарту NIST 800-30 | 28. Веб-платформа оцінки вразливості ПЗ |
| Cross-Site Scripting Guardian A Static XSS Detector [12] | 2020 | Метод статичного аналізу вихідного коду на вразливість для запобігання XSS-атаці із високою швидкістю виявлення | |
| A dynamic honeypot design for intrusion detection [13] | 2004 | Дизайн Honeypot як інструменту для вивчення методів, які використовують зловмисники в динамічному мережевому середовищі | |
| Intelligent system using machine learning techniques for security assessment and cyber intrusion detection [14] | 2022 | Інтелектуальна система, яка використовує методи ML для оцінки безпеки та виявлення кібервторгнень | |

Продовження табл. 8.2

| | | | |
|---|---|---|---|
| Machine learning approach to quick incident response [15] | 2020 | Підхід на основі ML для сортування рішень, які використовуються для швидкого реагування на кіберінциденти | |

Таблиця 8.3 – Стисла характеристика взятих до розгляду джерел, що описують використання вбудованих механізмів ШІ для пошуку, виявлення, класифікації і боротьби із атаками на систему під час її роботи

| 29. Назва | 30. Рік | 31. Що запропоновано | 32. Коментарі |
|---|---|---|---|
| 33. A Character-Level Neural Network Model for Web Attack Detection [16] | 34. 2019 | 35. Модель, яка поєднує методи згорткової нейронної мережі (Convolutional Neural Network (CNN)) та довготривалої короткочасної пам'яті. | 36. Забезпечується більша продуктивність у порівняння з випадками застосування тільки CNN, є можливість виявляти невідомі атаки. Запропоновано методологію застосування |
| A Machine Learning Approach to Malicious JavaScript Detection using Fixed Length Vector Representation [17] | 2018 | Підхід щодо виявлення шкідливого коду JavaScript (JS) | Наведено результати експериментів, здійснено порівняння продуктивності |
| A Novel Architecture for Web-Based Attack Detection Using [18] | 2020 | Архітектура виявлення веб-атак на основі аномалій | Представлені результати успішного виявлення атак |

Продовження табл. 8.3

| | | | |
|---|---|---|---|
| A Hybrid Intrusion Detection System Based on Scalable K-Means Random Forest and Deep Learning [19] | 2021 | Система виявлення вторгнень, яка використовує k-середні та алгоритми RF для двійкової класифікації звичайних подій і подій атаки. Потім за допомогою алгоритмів ШІ події, визнані ненормальними, класифікуються за різними типами атак | |
| An approach to detect user behaviour anomalies within identity federations [20] | 2021 | Підхід, який дозволяє визначити, коли відбувається значне відхилення від шаблонів або тенденцій, встановлених як стандарт для користувачів і організацій | |
| Architecture and Model of Neural Network Based Service for Choice of the Penetration Testing Tools [21] | 2021 | Модель нейронної мережі, яка на стороні сервера навчена на даних, отриманих від експертів у галузі тестування на проникнення. Рекомендації щодо вибору інструментів тестування на проникнення відповідно до заданих вимог | Показаний зразок |
| Data augmentation-based conditional Wasserstein generative adversarial network-gradient penalty for XSS attack detection system [22] | 2020 | Умовна генеративна змагальна мережа Вассерштейна з градієнтним штрафом для покращення можливостей системи виявлення XSS-атак у середовищі даних з низьким ресурсом | Експерименти на двох незбалансованих наборах даних XSS-атаки демонструють, що запропонована модель генерує дійсні та надійні зразки |

Продовження табл. 8.3

| | | | |
|---|---|---|---|
| Detecting web attacks using random undersampling and ensemble learners [23] | 2021 | Неповна вибірка та ансамбль учнів, порівняння різних класифікаторів для класифікації загроз | |
| Employing Machine Learning Techniques for Detection and Classification of Phishing Emails [24] | 2017 | Модель на основі нейронної мережі як інструмент для виявлення фішингових електронних листів | |
| Hunt for Unseen Intrusion: Multi-Head Self-Attention Neural Detector [25] | 2021 | Модель багатоголового самоконтролю (Multi-Head Self-Attention (MHSA)) як вдосконалений нейронний детектор для захоплення розрізнених фрагментів доказів у мережевому трафіку | |
| Phishing Websites Detection using Machine Learning [26] | 2019 | Система на основі машинного навчання для ідентифікації справжності веб-сайтів | |

### 8.3.1. Типи атак на веб-сервіси

Для того, щоб розглянути і дослідити методи боротьби із загрозами, потрібно мати розуміння про самі загрози, від яких планується захищатися. Різні автори мають відмінне бачення загроз і їх важливість. Автори [7] виділяють такі важливі типи атак, їхній вплив та особливості.

**Denial of Service (DoS) attacks**. DoS-атаки стають все більш витонченими і складнішими для виявлення, що пов’язано з легкою доступністю інструментів для зловмисників, а також поширенням ринку кіберзлочинності як послуги (Cyber Crime as a Service (CCaaS)).

**Phishing and spear-phishing attacks**. Такі атаки використовують принципи соціальної інженерії, коли підроблені електронні листи виглядають легітимними для кінцевих користувачів, що змушує останніх довіряти їм.

**Structured Query Language (SQL) injection attacks**. Такі атаки при виконанні на веб-сервері можуть розкрити частину або всі дані, що зберігається на сервері бази даних, зокрема, імена користувачів і їх паролі.

**Cross-site scripting**. Таке шкідливе ПЗ може передавати дані користувача з комп'ютера жертви на сервери зловмисника.

**Malware attacks**. У статті [14] відзначено, що такі атаки пошкоджують окремі важливі компоненти і роблять веб-сервіс непридатним для використання.

**Spamming**. Відповідно до [14] такі атаки досить легко виявляються.

У статті [28] описані декілька нових типів атак, що стали можливими із поширенням використання ШІ.

**Deepfake**. Ця технологія здатна створювати надзвичайно реалістичні зображення (наприклад, знаменитостей).

**Зламування CAPTCHA**. Сьогодні ML здатне зламувати CAPTCHA за 0,05 секунди, використовуючи генеративну змагальну мережу (Generative Adversarial Network (GAN)).

### 8.3.2. Використання технологій штучного інтелекту для протидії кіберзагрозам

Технології ШІ та їх використання для протидії кіберзагрозам розглядаються у декількох публікаціях: автори [28] виділили ключові особливості різних методів ШІ, а автори [1] описали особливості використання штучних нейронних мереж із прикладами.

На основі аналізу вище зазначених джерел [1, 28] переваги різних технологій ШІ показано у таблиці 4.

Таблиця 8.4 – Технології штучного інтелекту та їх переваги.

| 37. Технологія | 38. Переваги |
|---|---|
| 39. Штучні нейронні мережі | 40. Паралелізм в обробці інформації<br>41. Навчання на прикладі<br>42. Нелінійність – обробка складних нелінійних функцій<br>43. Стійкість до шуму та неповних даних<br>44. Універсальність і гнучкість моделей навчання |

Продовження табл. 8.4

| | |
|---|---|
| 45. Інтелектуальні агенти | 46. Мобільність<br>47. Раціональність у досягненні своїх цілей<br>48. Адаптивність до навколишнього середовища тавподобань користувача<br>49. Співпраця – усвідомлення того, що людина-користувач може робити помилки та надавати невизначену або пропускати важливу інформацію, тому агенти не повинні приймати інструкції без розгляду та перевірки невідповідностей з користувачем |
| 50. Генетичні алгоритми | 51. Міцність<br>52. Пристосованість до навколишнього середовища<br>53. Оптимізація – надання оптимальних рішень навіть для складних обчислювальних проблем<br>54. Паралелізм – дозволяє оцінювати декілька схем одночасно<br>55. Гнучкий і надійний глобальний пошук |
| 56. Нечіткі набори | 57. Надійність їх інтерполятивного механізму міркування<br>58. Інтероперабельність – доброзичливість до людини |

8.3.3. Використання застосунку на основі штучного інтелекту для аналізу і оцінки існуючих систем на вразливості

Розглянемо використання ШІ для аналізу вразливостей веб-сервісів. Метою статей, присвячених зазначеному питанню є запропонувати надійну інтелектуальну систему оцінювання безпеки на основі методів ML. Отже, головний інтерес полягає у визначенні ефективних механізмів пошуку вразливостей для захисту комп'ютерних мереж і систем. Такі механізми допомагають аналітикам виявляти вразливості та оцінювати їх. У зв'язку з цим значна частина статей присвячена виявленню найважливіших вразливостей системи, а також рішенням для їх уникнення. Крім того, у ряді джерел розглядаються особливості застосування методів веб-майнінгу для аналізування поведінки відвідувачів з метою виявлення їх аномальних дій.

**Аналіз вихідного коду.** Існують системи для статичного аналізу коду за допомогою ШІ замість людини [12], наприклад аналіз вихідного PHP коду на вразливості. Традиційний аудит коду зазвичай здійснюється вручну, що призводить до великого споживання людських ресурсів і високих показників

помилки. Таким чином, для допомоги аудиторам було розроблено кілька інструментів автоматичного аудиту. Однак велика кількість поточних інструментів керується великою бібліотекою правил і положень, що вказує на те, що їм не вистачає гнучкості. Зі зростанням попиту на аналіз даних, ефективне отримання знань за допомогою ML поступово стає основною рушійною силою. Однак, завжди постає питання: «Як виконати глибокий аналіз складних і різноманітних даних?». Наразі дослідження з векторизації коду дозволяють використовувати технологію ML в аудиті коду.

**Оцінювання вразливостей і їх важливості.** Крім того, також пропонуються системи для оцінювання вразливостей і ризиків кібербезпеки. Автори [11] запропонували програмну платформу під назвою «Управління вразливістю кіберризиків» (Cyber Risk Vulnerability Management (CYRVM)), засновану на стандарті NIST 800-30, яку можна використовувати для управління кіберризиками. Автори вирішили використати цю структуру, тому що вважають, що вона аналізує кіберризики, починаючи з вищого рівня й опускаючись до деталей. Найважливішими відмінними рисами цього ПЗ є те, що воно:

– є легко доступним для будь-якого веб-переглядача і не потребує жодної іншої інсталяції на комп'ютері великі дані – величезна кількість субдоменів та взаємозв'язків ускладнює систему та видає великий потік даних;

– є стандартизованим рішенням, оскільки відповідає NIST 800-30;

– здатне інтегрувати оцінку вразливості за допомогою імпорту звіту OpenVas (безкоштовна платформа з кількома службами та інструментами, що пропонують комплексне та потужне рішення для сканування вразливостей і керування вразливістю);

– здатне перераховувати всі вразливості для всіх активів, надаючи користувачеві повне розуміння з точки зору цілісності та конфіденційності;

– здатне розраховувати ймовірність події та оцінювати вплив на систему.

Дотримуючись процедур, описаних у стандарті NIST 800-30, ця програмна платформа може допомагати адміністратору мережі в аналізі ризиків. Представимо процес аналізу та оцінки кіберризиків у вигляді схеми, що складається із 9 кроків, графічно відображених на рисунку 1, а саме:

1. Характеристика системи. На цьому кроці визначаються межі системи разом із ресурсами та інформацією, що входить до неї.

2. Ідентифікація загроз. Метою цього кроку є ідентифікація потенційних джерел загроз і складання заяви про загрозу з переліком потенційних джерел загроз системі, що оцінюється.

3. Виявлення вразливостей. Метою цього кроку є отримання списку вразливостей системи (спостережень), які можуть використовуватися потенційними джерелами загроз.

4. Аналіз протидії. Метою цього кроку є аналіз засобів протидії, які були впроваджені або заплановані до впровадження організацією для мінімізації або усунення ймовірності того, що загроза спричинить уразливість системи.

5. Аналіз імовірності. Мета цього кроку полягає в тому, щоб отримати оцінку рейтингу ймовірності для кожної вразливості. Розглядається також мотивація та можливості джерела загрози, природа вразливості, поточні засоби контролю.

6. Аналіз впливу. Метою цього кроку є визначення негативного впливу в результаті успішного використання вразливості після отримання інформації про завдання системи (наприклад, виконувані процеси), небезпечність системи та даних (наприклад, значення або важливість системи для організації), конфіденційність системи та даних.

7. Визначення ризиків. Метою цього кроку є оцінка рівня ризику для системи.

8. Рекомендації з протидії. Метою цього кроку є запропонувати засоби протидії, які можуть пом'якшити або усунути ідентифіковані ризики відповідно до діяльності організації.

9. Документування результатів. Після завершення оцінки ризиків (виявлення джерел загроз і вразливостей, оцінка ризиків і надання рекомендованих засобів протидії) результати слід задокументувати в офіційному звіті або брифінгу.


Рисунок 8.1 – Схема процесу аналізу та оцінки кіберризиків

***Класифікація веб-сервісів.*** Тестування на проникнення є одним із способів виявлення проблем безпеки веб-застосунків. При проведенні такого тестування активно використовуються інструменти, призначені для автоматизації монотонних процесів. Проблема полягає в тому, що для тестування певних класів проблем безпеки веб-застосунків використовуються інструменти зі схожою функціональністю, і невідомо, який інструмент краще обрати для конкретного випадку. Тож автори [21] припустили, що ШІ може використовуватися для класифікації веб-сервісів, що тестуються для подальшого вибору найкращих інструментів саме для конкретного випадку.

***Переваги і недоліки.*** Перевагами зазначеного вище способу використання ШІ для забезпечення кібербезпеки веб-сервісів є те, що процедура оброблення і

пошуку вразливостей виконується один раз під час проектування, розроблення чи тестування та не сповільнює роботу системи. Крім того, ШІ не працює з реальними даними користувачів, оскільки він не вбудований до кінцевої системи, тож не може бути атакованим з метою викрадення чи підміни даних. Але такий спосіб використання ШІ має і недолік, який полягає у тому, що ШІ використовується для оцінювання захищеності системи лише на початку життєвого циклу ПЗ, а під час роботи системи участь у виявленні атак та протидії їм не приймає. Помилкові спрацювання можуть бути розглянуті людьми, а саме супервізорами та адміністраторами. Саме вони вирішують, чи правильно інструмент спрацював під час виявлення вразливості. Безумовно, із часом ШІ буде навчатися і його застосування призведе до зниження частоти помилкових спрацювань.

***Точність.*** Дуже важливим показником успішності застосування технологій ШІ та моделей на їх основі є точність. Точність спрацьовувань зростає із часом через навчання ШІ. Це є особливістю ШІ, крім того такий висновок можна зробити, якщо звернутися до графіків зміни точності спрацьовувань в залежності від кількості епох, що були використані під час навчання. Приклад такого графіку запропоновано на рисунку 8.2 [12].

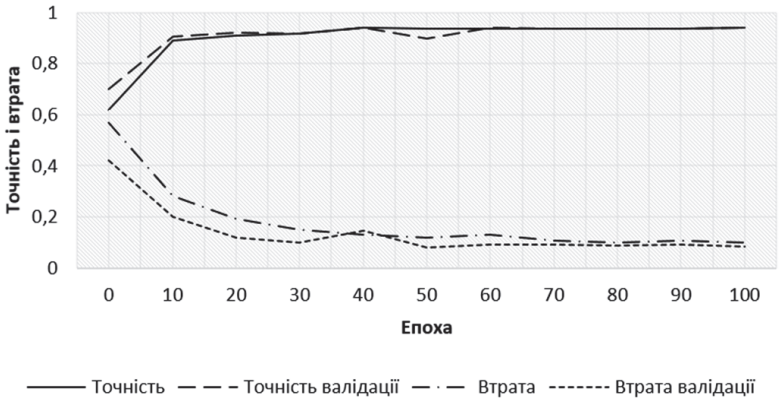



Рисунок 8.2 – Діаграма зміни точності при збільшенні кількості епох

Такий напрямок використання ШІ у кібербезпеці веб-сервісів є досить перспективним. Зокрема, у знайдених джерелах є статичний аналізатор PHP коду, тож можна було б розширити список мов, що підтримуються. Для інструментів, що класифікують веб-сервіси можна розширити список класів, і, відповідно, аналізаторів. Точність спрацьовування таких систем зростає із часом через використання ШІ.

8.3.4. Використання вбудованих механізмів штучного інтелекту для пошуку, виявлення, класифікації і боротьби із атаками на систему під час її роботи

Другим способом використання ШІ у кібербезпеці веб-сервісів є запровадження вбудованих механізмів і моделей для пошуку, виявлення, класифікації і протидії загрозам. Такі методи і моделі надають можливості із виявлення різного типу загроз і атак, класифікації типів атак. Крім того, існують методи протидії атакам. Для забезпечення функціональності веб-сервісів і безпеки даних користувачів було розроблено багато методів виявлення веб-атак. Через свою відкритість веб-сервіси завжди стають об'єктом багатьох різних типів атак. Для забезпечення функціональності веб-застосунків і безпеки даних користувачів було розроблено багато методів виявлення веб-атак. Такими методами є виявлення неправильного використання, виявлення аномалій або комбінація цих двох методів. Виявлення вторгнень добре працює на виявленні відомих атак, створюючи сигнатури атак і використовуючи їх для аналізу поведінки користувачів.

**Способи виявлення вторгнень.** В оглядовій статті [1] було виділено дві групи методів виявлення вторгнень: на основі сигнатур та на основі виявлення аномалій. Метод виявлення на основі сигнатур передбачає пошук неправильного використання або зіставлення шаблонів, які стають в нагоді при виявленні відомих атак, створюючи сигнатури для визначення поведінки користувачів. Метод виявлення на основі аномалій використовується для виявлення атак нульового дня та невідомих атак шляхом вивчення розташування та структури даних. Іншими словами, виявлення на основі аномалій – це тип виявлення вторгнень, який виявляє дані, відмінні від стандартного типу трафіку. Окрім того, автори [16] зазначають, що методи ML широко використовуються для автоматичного створення цих сигнатур атак шляхом навчання на позначених даних. Генетичний алгоритм (Genetic Algorithm (GA)) використовується для генерації сигнатур із виявлених атак. CNN використовується для аналізу URL-адрес для виявлення веб-атак і досягнення задовільного результату порівняно з моделлю n-грам і перевіркою людиною. І хоча ці методи ML для виявлення вторгнень можуть зменшити робоче навантаження на експертів із безпеки й виступати у ролі автоматизованого процесу для створення сигнатур атак, вони, однак, не демонструють здатності виявляти невідомі атаки.

***Naive Bayesian Classifier.*** Досить часто IDS генерує низку помилкових тривог, і ця проблема спонукала багатьох дослідників знайти рішення, щоб ідентифікувати сповіщення про менш важливий інцидент і зменшити помилкові тривоги, які є хибнопозитивними (False Positive (FP)) і хибнонегативними (False Negative (FN)). Так автори [27] запропонували IDS, засновану на техніці інтелектуального аналізу даних, яку можна використовувати для покращення роботи IDS у режимі реального часу, видалення нормальної активності з даних

тривоги для фокусування на реальних атаках і пошуку ненормальної активності, яка розкриває справжню атаку. Тобто відбувається обчислювальний процес виявлення закономірностей у наборах даних за допомогою методів ШІ та систем баз даних. Застосунки аналізу даних можуть використовувати різні параметри для дослідження різних наборів даних. Системи виявлення вторгнень у мережу (Network Intrusion Detection System (NIDS)) стали найважливішим компонентом сучасної мережевої інфраструктури через наслідки збільшення загроз безпеці в наш час. IDS генерує значну кількість тривог, проте в ній розгорнуті алгоритмічні процедури для зменшення кількості хибних спрацьовувань. На думку авторів дослідження [27] найбільш доцільним є використання для створюваної IDS двох основних технології – наївного байєсівського класифікатору (Naïve Bayes (NB)) і мультиноміальної логістичної регресії. Наївні байєсівські класифікатори можуть дуже ефективно навчатися в процесі керованого навчання. У багатьох практичних застосуваннях для оцінювання параметрів наївних байєсівських моделей застосовується метод максимальної правдоподібності. Мультиномінальна логістична регресія – це статистичний процес для представлення взаємозв'язків між змінними. Змінні, що використовуються для прогнозування інших змінних, називаються змінними-предикторами, а іноді незалежними змінними, тоді як змінні, які прогнозуються, називаються відгуком або залежною змінною. Регресійна модель називається простою регресією, коли є тільки одна змінна-предиктор, тоді як регресійна модель називається множинною регресією, якщо є більше однієї змінної-предиктора. Окрім того, запропонована авторами [27] система включає в себе методи майнінгу на двох послідовних рівнях, а саме: на першому рівні використовується наївний алгоритм Байєса для відокремлення аномальної активності від нормальної поведінки, а на другому – алгоритм мультиноміальної логістичної регресії для класифікації аномальної активності на основні чотири типи атак на додачу до нормального класу. Під час свого дослідження автори дослідили можливості деяких IDS, виконали тестування різних підходів і наприкінці свого дослідження зробили порівняння свого запропонованого підходу до створення IDS із існуючими. Результати тестування були зведені у порівняльну таблицю 5.

Таблиця 8.5 – Порівняння точності використовуваних методів машинного навчання

| 59. Джерело | 60. Назва джерела | 61. Використаний метод | 62. Точність | 63. Точність запропонованої системи | |
|---|---|---|---|---|---|
| | | | | 64. Рівень 1 | 65. Рівень 2 |

Продовження табл. 8.5

| | | | | | |
|---|---|---|---|---|---|
| 66. [29] | 67. | 68. Логісти- | 69. 87.7% | 70. 0 | 71. 97% |

| | | | | | |
|---|---|---|---|---|---|
| | Performance Evaluation of Intrusion Detection System Using Classification Algorithms | чна регресія | | | |
| | | 72. Наївний байєсовський | 73. 81.8% | 74. 98 % | 75. 0 |
| 76. [30] | 77. Network Intrusion Detecti-on System Using various data mining techniques | 78. Логістична регресія | 79. 80% | 80. 0 | 81. 97% |
| 82. [31] | 83. Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection | 84. Логістична регресія | 85. 84% | 86. 0 | 87. 98% |
| | | 88. Наївний байєсовський | 89. 79% | 90. 98 % | 91. 0 |

Проаналізувавши отримані дослідниками результати можна зробити висновок, що запропонована система дає перевагу у точності перед стандартними рішеннями.

**Класифікація атак.** Робота [19] пропонує систему для виявлення та класифікації атак. Принцип роботи такого рішення розділений на три етапи. Перший етап передбачає обробку вихідного набору даних виявлення вторгнень, включаючи цифрову обробку категоріальних функцій, обробку нормалізації цифрових функцій і видалення непотрібних даних. Другий етап полягає в класифікації нормальних і аномальних подій на основі комбінації k-середніх і RF на Spark. На третьому етапі приховані особливості вивчаються за допомогою алгоритмів глибокого навчання і виконується класифікація виявлених атак за типами. На виході ми отримуємо знайдені атаки, причому кожній із них присвоюється тип.

***XSS (Cross-site-scripting).*** Для такого типу атак пропонується окрема система, тому що цей тип атак є дуже небезпечним. Шкідливий JS-код може використовуватися для запису натискань клавіш, крадіжки файлів cookie браузера, зламу, пошкодження веб-сторінки, троянських коней тощо. Крім того, можна створити ботнет, змусивши користувачів завантажувати зловмисне ПЗ за допомогою соціальної інженерії.

Звичайні рішення безпеки, такі як антивірус і IDS, використовують підходи на основі підпису та евристики для виявлення атак. Евристичні підходи, такі як аналіз файлів, емуляція файлів і загальне виявлення сигнатур, базуються на наборі експертних правил прийняття рішень. Вони можуть виявляти раніше невідомі варіанти шкідливих програм.

Однак ці підходи мають низку недоліків. По-перше, знання про майбутню атаку є важливим для оновлення програми для ефективного виявлення, що створює вікно вразливості, яке зловмисники можуть використовувати для запуску нульового дня та інших типів атак. По-друге, тривалий час аналізу та сканування в поєднанні з великою кількістю помилкових спрацьовувань знижує продуктивність. Інші звичайні підходи до виявлення зловмисного JS – це використання шаблонів, приманки клієнта з низьким і високим рівнем взаємодії та підтримка чорного списку URL-адрес.

Автори [17] зазначають, що останнім часом було запропоновано ряд інтелектуальних систем виявлення вторгнень, які в основному зосереджені на виявленні аномалій. Сфери застосування ML у теперішній час надзвичайно розширюються, включаючи такі програми, як класифікація тексту та кластеризація, де модель ML використовується для таких завдань, як пошук документів, аналіз настроїв, фільтрація спаму, веб-пошук тощо. Текстові документи використовуються для навчання моделі ML, яка, у свою чергу, використовується для прогнозування класів або категорій документів. Модель ML вчиться на основі даних і може дати точний прогноз для заданих вхідних даних, якщо навчання буде проведено належним чином. Щоб досягти високопродуктивного прогнозування, необроблені дані зазвичай потрібно перетворити на належне представлення функцій для вхідних даних. Одним із таких представлень є вектор фіксованої довжини для текстових документів. Ці

моделі досягли найсучасніших результатів для класифікації тексту та завдань кластеризації на різних наборах даних.

Таким чином, оскільки JS-код містить текстові дані, передбачається, що можна класифікувати такі коди за допомогою моделей ML. Зокрема, можна вивчати функції шкідливого вмісту JS-коду за допомогою Doc2Vec і класифікатора для класифікації векторів функцій вмісту JS-коду.

**Web Application Firewall (WAF).** Брандмауер веб-застосунків (Web Application Firewall (WAF)) діє як бар'єр між веб-застосунком і клієнтом в Інтернеті, коли він розгорнутий перед веб-застосунком. Автори [32] відмічають, що окрім набору правил, відомих як політики, якими керується WAF, доцільно використовувати попередньо навчений модуль для прогнозування нових вхідних запитів. Використовуючи довготривалу короткочасну пам'ять (Long Short-Term Memory (LSTM)) як підхід до глибокого навчання, запропонована авторами [32] модель під час дослідження виявляла DDoS, XSS та SQL-ін'єкції з досить високою точністю.

Автори [33] проаналізували та реалізували модель глибокого навчання з шаром LSTM та вважають доцільним використання наборів запитів протоколу передачі гіпертексту (HyperText Transfer Protocol (HTTP)) та відповідей для навчання моделі виявлення загроз.

У статті [34] автори запропонували модель WAF, яка використовує методи машинного навчання та інженерії ознак для виявлення поширених веб-атак. Вони стверджують, що врахували основні обмеження попередніх робіт (невикористання заголовків запитів, використання лише одного набору даних, відсутність загальних ознак). Наведені ними результати свідчать про більшу точність роботи запропонованих моделей у порівнянні з існуючими.

Однак оптимальним є поєднання звичайних та інтелектуальних методів запобігання атакам. Так у [35] у розглянута гібридна модель WAF, яка для запобігання веб-атакам використовує методи виявлення як на основі сигнатур (SBD), так і на основі аномалій (ABD). Виявлення відомих веб-атак здійснюється за допомогою SBD, а виявлення аномальних HTTP-запитів – за допомогою ABD. ABD на основі навчання реалізується за допомогою штучних нейронних мереж (Artificial Neural Network (ANN)). За результатами тестування було отримано високий середній відсоток успішності (96,59 %).

**Переваги і недоліки.** Застосування ШІ під час роботи системи, обробки запитів, роботи з даними, аналізі поведінки користувачів надає можливість виявляти атаки і аномальну поведінку в режимі реального часу, а також дозволяє класифікувати атаки за типами, протидіяти атакам та повідомляти адміністратора про небезпеку. Навчання ШІ на реальних даних сприяє підвищенню якості реагування на кібератаки. До недоліків можна віднести те, що ШІ сам може стати об'єктом атаки. Зокрема дані, на яких він навчається, можуть бути спотворені і це призведе до неправильних результатів. Також відзначимо, що застосування ШІ створює додаткове навантаження на апаратні

потужності обладнання, на якому працює веб-сервіс, а для подальшого навчання і підвищення точності (особливо під час виникнення спірних ситуацій) необхідно використовувати супервізор.

### 8.4. Варіанти реалізації

Під час аналізу існуючих публікацій було виявлено, що існує досить багато різноманітних аналізаторів вихідного коду, засобів для тестування на проникнення, тощо. Крім того вистачає і вбудованих технологій і механізмів, а також можна використовувати брандмауер. На нашу думку, користувачу чи розробнику при створенні веб-сервісу через величезну кількість засобів стає досить довго та важко підібрати правильні засоби для кожного розроблюваного продукту. Через це було вирішено запропонувати декілька методів із полегшення процесу пошуку, оцінки та виправлення вразливостей:

- механізм, заснований на стандарті NIST 800-30, що проведе користувача через всі етапи виявлення, оцінки та виправлення вразливостей;
- інструмент, який шляхом аналізу інформації про реалізацію, використані технології та вимоги до веб-сервісу буде надавати рекомендації, що описують, які саме популярні інструменти аналізу захищеності найкраще підходять та які механізми забезпечення кібербезпеки слід додати.

#### 8.4.1. Процес аналізу, заснований на NIST 800-30

На основі інформації, отриманої під час аналізу методів забезпечення кібербезпеки веб-сервісів і існуючих рішень дійшли висновку про необхідність розроблення заснованого на стандарті NIST 800-30 методу і веб-застосунку для аналізу захищеності та управління кіберризиками.

NIST SP 800-30 "Guide for Conducting Risk Assessments" ("Посібник із проведення оцінок ризику") присвячений процедурі проведення оцінювання ризику.

Дотримуючись процедур, описаних у стандарті NIST 800-30, розроблювані метод та веб-застосунок зможуть допомагати в аналізі кіберзахищеності і ризиків. ШІ планується використовувати на багатьох етапах процесу аналізу для надання кращих рекомендацій і допомоги користувачу. До переваг розроблюваного веб-застосунку слід віднести відсутність необхідності у його інсталяції на комп’ютері користувача та доступність через веб-браузер.

#### 8.4.2. Метод і інструмент для полегшення вибору найефективніших засобів і механізмів для виявлення вразливостей

Через величезну кількість доступних на ринку інструментів було зроблено висновок, що ефективні механізми виявлення вразливостей існують у достатній кількості, і створювати ще один буде не дуже ефективно. Тому доцільним буде запропонувати розроблення методу і інструменту для полегшення вибору найефективніших засобів і механізмів для кожного конкретного випадку. На основі вхідних даних про досліджуваний веб-сервіс створюваний інструмент повинен надавати рекомендації з використання певних засобів для подальшого пошуку вразливостей та тестування. Окрім того, він повинен мати можливість надавати рекомендації про те, які саме механізми для забезпечення кібербезпеки слід використати.

У створенні інструменту доцільним може бути використання технологій ШІ для підвищення точності рекомендацій.

Створюваний інструмент також може бути виконаним у вигляді веб-застосунку.

## 8.5. Висновки

Існують два основні підходи, що можуть бути затребуваними для забезпечення кібербезпеки веб-сервісів, а саме: використання застосунків на основі ШІ для аналізу і оцінки веб-сервісу на вразливості; використання вбудованих механізмів ШІ для пошуку, виявлення, класифікації і боротьби з атаками на веб-сервіс під час його роботи.

Для забезпечення кібербезпеки веб-сервісів доцільно застосовувати комбінування конкретного етапу захисту і конкретної технології ШІ. Треба також брати до уваги той факт, що технологія ШІ, успішно застосовувана для виявлення одного типу атак, може виявитися зовсім неефективної під час виявлення атак іншого типу.

Показано, що застосування досліджуваних методів може супроводжуватися хибно-позитивними та хибно-негативними спрацьовуваннями. Підвищення точності роботи ШІ може бути забезпечено шляхом використання для здійснення навчання супервізора, здатного відслідковувати правильні та неправильні спрацювання.

Використання ШІ для забезпечення кібербезпеки веб-сервісів несе ризики і обмеження, що полягають у зниженні швидкодії або додатковому навантаженні на обчислювальні потужності підсистем, відповідальних за застосування ШІ.

Через величезну кількість існуючих засобів і механізмів для виявлення вразливостей доцільним буде запропонувати розроблення методу і інструменту для полегшення вибору найефективніших із таких засобів і механізмів для кожного конкретного випадку.

Література

1. Sattikar A.A., Kulkarni R.V., A Role of Artificial Intelligence Techniques in Security and Privacy Issues of Social Networking (2012), International Journal of Computer Science Engineering and Technology, Vol. 2, No 1, pp. 792–795.

2. Ishaq A.M., Artificial intelligence for cybersecurity: a systematic mapping of literature (2020), IEEE Access, Vol. 8, No 1, pp. 172–176. DOI: 10.1109/ACCESS.2020.3013145.

3. Das R., Sandhane R., Artificial Intelligence in Cyber Security (2021), Journal of Physics: Conference Series, Vol. 1964, article 042072. DOI: 10.1088/1742-6596/1964/4/042072.

4. Abdullahi M., Baashar Y., Alhussian H., Alwadain A., Aziz N., Capretz L.F., Abdulkadir S.J., Detecting Cybersecurity Attacks in Internet of Things Using Artificial Intelligence Methods : A Systematic Literature Review (2022), Electronics, Vol. 11, No 2, article 198. DOI: 10.3390/electronics11020198.

5. Kirichenko L., Radivilova T., Anders C., Detecting cyber threats through social network analysis: short survey (2017), SocioEconomic Challenges, Vol. 1, No 1, pp. 20–34. DOI: 10.21272/sec.2017.1-03.

6. Radwan M., Tariq K., Difficulties Faced and Applications of Machine Learning in Cyber-Security (2021), International Journal of Advances in Soft Computing and its Applications, Vol. 13, No 2, pp. 162–172.

7. Zeadally S., Adi E., Baig Z., Khan I., Harnessing Artificial Intelligence Capabilities to Improve Cybersecurity (2020), IEEE Access, Vol. 8, pp. 23817–23837. DOI: 10.1109/ACCESS.2020.2968045.

8. Faesen L., Frinking E., Gricius G., Mayhew E., Understanding the Strategic and Technical Significance of Technology for Security (2019), The Hague Security Delta (HSD), Den Haag, Nederland, Available at: https://hcss.nl/wp-content/uploads/2021/01/HSD-Rapport-Data-Diodes.pdf (accessed August, 2022).

9. Musser M., Garriott A., Machine Learning and Cybersecurity: Hype and Reality (2021), Center for Security and Emerging Technology, Washington, USA. Available at: https://cset.georgetown.edu/wp-content/uploads/Machine-Learning-and-Cybersecurity.pdf.

10. Jemal I., Haddar M.A., Cheikhrouhou O., Mahfoudhi A., Performance evaluation of Convolutional Neural Network for web security (2021), Computer Communications, Vol. 175, pp. 58–67. DOI: 10.1016/j.comcom.2021.04.029.

11. Russo P., Caponi A., Leuti M., Bianchi G. A., Web Platform for Integrated Vulnerability Assessment and Cyber Risk Management (2019), Information, Vol. 10, No 7, article 242. DOI: 10.3390/info10070242

12. Li C., Wang Y., Miao C., Huang C., Cross-Site Scripting Guardian: A Static XSS Detector Based on Data Stream Input-Output Association Mining (2020), Applied Sciences, Vol. 10, No 14, article 4740. DOI: 10.3390/ app10144740

13. Kuwatly I., Sraj M., Masri Z.A., Artail H. A., dynamic honeypot design for intrusion detection (2004), Proceedings of the IEEE/ACS International Conference on Pervasive Services (ICPS), pp. 95–104. DOI: 10.1109/perser.2004.3.

14. Abdel K., Intelligent system using machine learning techniques for security assessment and cyber intrusion detection (2022), Angers: Université d'Angers. Available at: https://theses.hal.science/tel-03522384/file/KASSEM.pdf.

15. Nila C., Apostol I., Patriciu V., Machine learning approach to quick incident response (2020), Proceedings of the13th International Conference on Communications (COMM), 2020, pp. 291–296. DOI: 10.1109/COMM48946.2020.9141989.

16. Gong X., Lu J., Wang Y., Qiu H., He R., Qiu M., CECoR-Net: A Character-Level Neural Network Model for Web Attack Detection (2019), Proceedings of the 4th IEEE International Conference on Smart Cloud, SmartCloud 2019 and 3rd International Symposium on Reinforcement Learning (ISRL), pp. 98–103. DOI: 10.1109/SmartCloud.2019.00027.

17. Ndichu S., Ozawa S., Misu T., Okada K., A Machine Learning Approach to Malicious JavaScript Detection using Fixed Length Vector Representation (2018), Proceedings of the International Joint Conference on Neural Networks (IJCNN). DOI: 10.1109/IJCNN.2018.8489414.

18. Tekerek A., A novel architecture for web-based attack detection using convolutional neural network (2021), Computers and Security, Vol. 100. DOI: 10.1016/j.cose.2020.102096.

19. Liu C., Gu Z., Wang J., A Hybrid Intrusion Detection System Based on Scalable K-Means+ Random Forest and Deep Learning (2021), IEEE Access, Vol. 9, pp. 75729–75740. DOI: 10.1109/ACCESS.2021.3082147.

20. Martín A.G., Beltrán M., Fernández-Isabel A., Martín de Diego I., An approach to detect user behaviour anomalies within identity federations (2021), Computers and Security, Vol. 108. DOI: 10.1016/j.cose.2021.102356.

21. Tetskyi A., Kharchenko V., Uzun D., Nechausov A., Architecture and Model of Neural Network Based Service for Choice of the Penetration Testing Tools (2021), International Journal of Computing, Vol. 20, No 4, pp. 513–518. DOI: 10.47839/ijc.20.4.2438.

22. Mokbal F.M.M., Wang D., Wang X., Fu L., Data augmentation-based conditional Wasserstein generative adversarial network-gradient penalty for XSS attack detection system (2020), PeerJ Computer Science, Vol. 6, pp. 1–20. DOI: 10.7717/peerj-cs.328.

23. Zuech R., Hancock J., Khoshgoftaar T.M., Detecting web attacks using random undersampling and ensemble learners (2021), Journal of Big Data, Vol. 8, No 1. DOI: 10.1186/s40537-021-00460-8.

24. Moradpoor N., Clavie B., Buchanan B., Employing machine learning techniques for detection and classification of phishing emails (2018), Proceedings of the 2017 Computing Conference, pp. 149–156. DOI: 10.1109/SAI.2017.8252096.

25. Seo S., Han S., Park J., Shim S., Ryu H.E., Cho B., Lee S., Hunt for Unseen Intrusion: Multi-Head Self-Attention Neural Detector (2021), IEEE Access, Vol. 9, pp. 129635–129647. DOI: 10.1109/ACCESS.2021.3113124.

26. Kiruthiga R., Akila D., Phishing websites detection using machine learning (2019), International Journal of Recent Technology and Engineering, Vol. 8, No 2, Special Issue 11, pp. 111–114. DOI: 10.35940/ijrte.B1018.0982S1119.

27. Shareef S., Hashim S., Proposed Hybrid Classifier to Improve Network Intrusion Detection System using Data Mining Techniques (2020), Engineering and Technology Journal, Vol. 38, No 1B, pp. 6–14. DOI: 10.30684/etj.v38i1b.149.

28. Pupillo L., Fantin S., Ferreira A., Polito C., Final Report of a CEPS Task Force on Artificial Intelligence and Cybersecurity (2021), Brussels: Centre for European Policy Studies (CEPS), 122 p. Available at: https://www.ceps.eu/wp-content/uploads/2021/05/CEPS-TFR-Artificial-Intelligence-and-Cybersecurity.pdf.

29. Manju C., Performance evaluation of intrusion detection system using classification algorithms (2017), International Journal of Innovative Research in Science, Engineering and Technology, Vol. 6, No 7, pp. 15051–15057. DOI:10.15680/IJIRSET.2017.0607329.

30. Gupta D., Singhal S., Malik S., Singh A., Network intrusion detection system using various data mining techniques (2016), Proceedings of the International Conference on Research Advances in Integrated Navigation Systems (RAINS). DOI: 10.1109/RAINS.2016.7764418.

31. Belavagi M.C., Muniyal B., Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection (2016), Procedia Computer Science, Vol. 89, pp. 117–123. DOI: 10.1016/j.procs.2016.06.016.

32. Dawadi B.R., Adhikari B., Srivastava D.K., Deep Learning Technique-Enabled Web Application Firewall for the Detection of Web Attacks (2023), Sensors, Vol. 23, No 4, article 2073. DOI: 10.3390/s23042073.

33. Toprak S., Yavuz A.G., Web application firewall based on anomaly detection using deep learning (2022), Acta Infologica, Vol. 6, No 2, pp. 219–244. DOI: 10.26650/acin.1039042.

34. Aref S., Bassam Kurdy M.H.D., Web Application Firewall Using Machine Learning and Features Engineering (2022), Security and Communication Networks, Vol. 2022, article 5280158. DOI: 10.1155/2022/5280158.

35. Tekerek A., Bay O.F., Design and implementation of artificial intelligence-based web application firewall model (2019), Neural Network World, Vol. 29, No. 4, pp. 189–206. DOI: 10.14311/NNW.2019.29.013.

# 9. РОЗРОБКА МОДЕЛІ ЗАГРОЗ ДЛЯ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

**Г. А. Землянко[1], В. Я. Пєвнєв[1], Ніколаос Бардіс[2], В. С. Харченко[1]**

*[1]Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»*
*[2]Hellenic Army Academy, Греція*

## 9.1. Вступ

Нині використання безпілотних літальних апаратів (БПЛА) стає дедалі поширенішим у різних галузях, починаючи від військової сфери і закінчуючи застосуванням у комерційних цілях. Однак у зв'язку зі збільшеною популярністю цих систем, зростає і ризик виникнення кібератак та інших загроз, пов'язаних із кібербезпекою.

Нині в різних сферах діяльності ставляться завдання, які можуть бути вирішені за допомогою БПЛА, організованих у "рій". Під роєм БПЛА розуміють самоорганізовану систему, елементи якої спілкуються між собою і на основі цього можуть шукати колективно-вироблені рішення для вирішення конкретної задачі [1]. У контексті цієї роботи буде розглянуто, як один із варіантів, рішення про виключення агентів, що несуть загрозу для функціонування рою, з інформаційної взаємодії (ІВ) за відсутності людського чинника, тобто на основі інформації, що передається між агентами. Порушення інформаційної безпеки (ІБ) рою БПЛА можуть призвести до втрат працездатності цього рою. Таким чином, автори розглядають питання аналізу інформаційної взаємодії, особливостей наявних вразливостей та наявних загроз як одне з основних питань з точки зору досягнення цілей, поставлених перед роєм.

У зв'язку з популяризацією БПЛА в різних цивільних галузях підвищується ймовірність виникнення ситуацій порушення працездатності рою БПЛА через різні загрози. Як правило, на всіх етапах розроблення робототехнічної системи фактору захищеності системи приділяється недостатня увага, у зв'язку з чим така система під час функціонування схильна до різних інформаційних загроз [2]. У зв'язку з цим одним із найважливіших завдань цього дослідження є забезпечення захищеного групового функціонування БПЛА.

Слід виділити роботу [3], у якій розв'язується проблема забезпечення ІБ у рої БПЛА. Об'єктом дослідження є безпілотний авіаційний комплекс розвідки, який функціонує спільно з наземним керуючим комплексом. Дослідження показало вразливість каналу зв'язку (КЗ), а також обчислювального центру БПЛА для актуальних кіберфізичних загроз. Методи, використані в роботі, розраховані на централізовані стратегії роєвого управління, отже, вони не підходять для аналізу децентралізовано-організованих систем.

Автори роботи [4] розглядають рій БПЛА заснований на мультиагентному підході. Це дослідження націлене на теоретичне обґрунтування необхідності організації захисту рою БПЛА від актуальних атак.

У роботах [5, 6] розглядаються рої БПЛА, що використовують децентралізовані стратегії при взаємодії між БПЛА, але мають наземні центри управління, що означає впровадження змішаної стратегії роєвого управління. Аналіз цього матеріалу показав необхідність впровадження методів захисту ІВ. Незважаючи на успішні досягнення в галузі забезпечення ІБ ІВ, використання людського ресурсу та центрів управління підвищує ризик порушення функціонування роїв.

Нині БПЛА середнього та важкого класу застосовують для розв'язання широкого спектра завдань, таких як патрулювання кордонів, розвідки, транспортування та збройних атак.

Такі події як захоплення RQ-170 Sentinel Збройними силами Ірану 4 грудня 2011 року [7] або вірус-шпигун, який інфікував флот американських безпілотників на авіабазі ВПС США Кріч (Creech Air Force Base) у Неваді у вересні 2011, [8] доводять недостатність захисту БПЛА.

У цій роботі розглядається проблема вразливості каналів управління БПЛА, виходячи з можливих загроз, які можуть бути використані зловмисниками. Особлива увага приділяється розробці моделі загроз для безпілотних систем, яка дозволить забезпечити ефективний захист від кібератак та інших загроз.

Актуальність даної теми полягає у тому, що БПЛА застосовуються в різних галузях, включаючи комерційні цілі, такі як доставка, моніторинг та перевезення вантажів, а також у військових операціях, телекомунікаційних мережах та інших сферах. За даними 2012 року, військова система США збільшила свої інвестиції у дослідження та виробництво БПЛА з 2,3 млрд. доларів у 2008 році до 4,2 млрд. доларів [9]. До 2022 року планується подальше збільшення цього обсягу до 8,6 млрд. доларів [10].гру Однак, поряд з перевагами, такими як підвищення продуктивності та зниження ризику для людини, БПЛА також стають об'єктом інтересу для зловмисників, які можуть використовувати їх для скоєння злочинів, шпигунства або терористичних актів, серед іншого.

Метою даної роботи є вивчення існуючих вразливостей каналів керування БПЛА та розробка моделі загроз для безпілотних систем, що дасть змогу визначити потенційні загрози та розробити заходи для їх запобігання. Для досягнення цієї мети требо виконати наступні завдання:

- дослідити наявні вразливості каналів управління БПЛА та можливі загрози, пов'язані з кібербезпекою;
- розробити модель загроз для безпілотних систем, враховуючи особливості їхньої роботи та можливості зловмисників.

## 9.2 Визначення загроз для безпеки каналів управління БПЛА

В Україні застосування БПЛА регулюється законодавством та нормативними документами, які включають в себе вимоги до кваліфікації операторів, обмеження на місцях польотів, вимоги до технічного стану БПЛА, умовами використання та інші.

Основним нормативним документом, що регулює застосування БПЛА в Україні, є "Правила ведення робіт з використанням безпілотних літальних апаратів" (Наказ Державної служби України з екології та природних ресурсів №616 від 21.07.2015) [11]. Ці правила містять загальні вимоги до застосування БПЛА, а також визначають порядок отримання дозволів на їх використання.

Крім того, існують стандарти, які можуть застосовуватися під час використання БПЛА, такі як "Авіаційні правила України. Частина XXIII. Безпілотні літальні апарати" (ДСТУ-3745:2018) [12], а також "Безпілотні аеророзвідувальні системи. Технічні вимоги" (ДСТУ EN 16605:2015) [13].

Загалом, застосування БПЛА в Україні регулюється з урахуванням міжнародних стандартів і рекомендацій, таких як Міжнародні стандарти цивільної авіації (ICAO) [14] і рекомендації Європейського агентства авіаційної безпеки (EASA) [15].

В Україні наразі не існує будь-яких вимог і стандартів, що регламентують систему управління для середніх і важких БПЛА, у зв'язку з чим для синтезу системи управління використовується система стандартів НАТО. Схему БПЛА в країнах НАТО, визначено в цій роботі [16].

Згідно з [8] БПЛА має складатися з трьох основних елементів: системи польоту БПЛА (air vehicle element), цільового навантаження (Payload element) і системи управління (UAV air component). Для аналізу можливості зовнішнього впливу буде розглянуто елементи, які можуть взаємодіяти з іншими компонентами за допомогою бездротової лінії зв'язку (радіо, оптичної, акустичної). У цьому випадку це може бути система управління і цільове навантаження.

Провівши аналіз схеми організації зв'язку системи управління БПЛА, можна припустити можливість наявності трьох векторів впливу на систему:

- центр управління польотом БПЛА;
- БПЛА;
- радіоканал БПЛА.

Вплив на центр управління польотом можна здійснювати з двох сторін. Перший вплив з боку зовнішніх мереж передавання даних, шляхом обходу захисту і подальших дій шкідливого впливу (впровадження програмних закладок, переспрямування трафіку з подальшою підміною пакетів і команд керування, а також інших впливів).

Переваги:

- повний контроль над БПЛА з функціоналом оператора;

- імовірнісна можливість впливу на інші підпорядковані центру управління польотом БПЛА складові частини системи управління.

Недоліком підходу стає величезна кількість часто непереборних факторів:

- для реалізації підходу необхідна висока кваліфікація фахівців, які здійснюють перехоплення управління;
- необхідна велика кількість знань конфіденційного характеру (потрібна довга розвідка архітектури мережі, протоколів взаємодії та багато іншої інформації);
- необхідний доступ до зовнішніх мереж передавання даних, що мають з'єднання з мережами технічного обслуговування БПЛА;
- результат взаємодії не визначений, тому що на нього впливає багато імовірнісних чинників.

Другим вектором взаємодії на центр управління БПЛА є нав'язування оператору неправдивої інформації щодо стану БПЛА і його просторового положення, через апаратуру приймання-передавання команд і відправлення телеметрії або приймання даних від цільового навантаження, шляхом підміни трафіку, який надходить на вхідний тракт приймального пристрою центру управління: стан об'єкта БПЛА (швидкості, кута атаки, висоти та стану інших датчиків), передання неправдивих даних із метою провокації оператора на дії, потрібні атакуючому та інші впливи. атакуючому, та інші впливи.

Переваги:

- порівняно нескладна реалізація;
- є реальна можливість спровокувати оператора на дії, необхідні атакуючому;
- збій польотного завдання.

Недоліки:

- необхідно перебувати в прямій радіовидимості антен апаратури приймання-передавання даних центру управління польотів;
- немає повного контролю над БПЛА;
- реалізація впливу і його наслідки сильно залежать від досвіду оператора;
- для реалізації атаки необхідний діяльний аналіз протоколів зв'язку БПЛА з ЦУП, що може бути ускладнено;
- канал передавання даних телеметрії і даних із цільового навантаження може бути захищений криптографічно (імітовставка або шифрування), що зводить впливи нанівець.

Вплив на БПЛА можна здійснити з трьох напрямків:

***Вплив на БПЛА шляхом нав'язування приймачу хибного (завищеного) значення сигнал/шум***

Сигнал/шум (S/N) - це відношення потужності сигналу до потужності шуму, присутнього в каналі зв'язку. Нав'язування неправильних значень сигналу/шуму може відбуватися з-за різних причин, таких як поміхи від інших радіоісточників, кібератаки або злонамерені дії.

Захоплення керування шляхом введення хибного (завищеного) значення сигнал/шум на вході приймального тракту може спричинити зниження чутливості приймача та перешкоджати сприйманню команд оператора. Це може призвести до перехоплення керування системою.

Перевагою методики є можливість здійснити повне захоплення управління БПЛА.

Недоліки:

- необхідність знання протоколів зв'язку;
- як і в разі впливу на оператора наявність системи криптозахисту інформації (СКЗІ) в каналі зведе до мінімуму ймовірність перехоплення управління;
- деякі системи зв'язку можуть бути несприйнятливими до цього виду атаки.

***Вплив на цільове навантаження***

Цей вид впливу можливий у тих випадках, коли відмова або неправильне функціонування цільового навантаження призводить до негайної зміни або припинення польотного завдання. Ефективним може стати вплив через радіоканали на цільове навантаження. Особливості реалізації впливу залежать від конкретного типу цільового навантаження.

Перевагою такого впливу є те, що в деяких випадках він може призвести до зриву польотного завдання.

Недоліки:

- не може або дуже рідко може призвести до захоплення управління БПЛА;
- використання СКЗІ зведе нанівець ефект від впливу.

***Вплив на систему просторового позиціонування БПЛА***

Вплив на систему просторового позиціонування БПЛА відкриває великі можливості для комбінування різних дій на зовнішні датчики БПЛА.

Цей вплив найефективніший у момент керування за допомогою автопілота або активного радіозаглушення каналів управління БПЛА і може призвести до часткового або повного перехоплення управління БПЛА, зриву польотного завдання або переведення системи в невизначений стан.

Також важливою особливістю впливу на систему просторового позиціонування є той факт, що спочатку подібні системи розроблялися для пілотованої авіації і в самій ідеї їхньої побудови не поставало так гостро питання протидії цілеспрямованому зловмисному впливу. У зв'язку з викладеним, ймовірність виявлення вразливості набагато вища, ніж в інших методах впливу.

Переваги:

- може призвести до часткового або повного захоплення управління;
- порівняно легка реалізація;
- вплив погано детектується з боку оператора, особливо, якщо його здійснюють у режимі автоматичного пілотування;

- для протидії необхідне розроблення нових виробів.

Недоліки

- потрібна попередня розвідка апаратури просторового позиціонування;
- через широкий спектр діапазону впливу необхідно мати велику кількість різного обладнання, що працює на різних частотних діапазонах, що не завжди є можливим.

Вплив на ретранслятори зв'язку в цій роботі не розглядались, так як, є залежність можливого впливу від конкретної реалізації.

Вплив на систему просторового позиціонування БПЛА є однією з найбільш серйозних загроз безпеці при використанні безпілотних літальних апаратів. Це може призвести до втрати контролю над апаратом та його падіння, що може завдати шкоди як майнові, так і людській безпеці.

У зв'язку з тим, що системи управління БПЛА були розроблені з огляду на забезпечення їх надійності та високої продуктивності, забезпечення безпеки управління БПЛА залишалося на другому плані. Проте, з огляду на зростання кількості застосування БПЛА та збільшення кількості випадків їх використання для злочинних дій, питання безпеки управління БПЛА набуває все більшої актуальності.

В результаті проведеного аналізу було виявлено перелік векторів впливу на систему управління БПЛА, які можуть стати об'єктом зловмисних дій. Цей перелік представлений в таблиці 9.1. До основних векторів впливу на систему управління БПЛА відносяться перешкоджання зв'язку між земною станцією та БПЛА, вплив на датчики апарату з метою їх збоїв або випадкових вимірювань, введення шумів в систему передачі даних, а також атаки на програмне забезпечення БПЛА.

Таблиця 9.1 – Вектори впливу на систему управління БПЛА

| Вектор впливу | Переваги | Недоліки |
|---|---|---|
| Впливи на центр управління БПЛА з боку зовнішніх мереж. | 1 Повний контроль над БПЛА з функціоналом оператора.<br>2. Імовірнісна можливість впливу на інші підпорядковані центру управління польотом БПЛА складові частини системи управління. | 1. Складна реалізація, автоматизація.<br>2. Необхідна велика кількість знань конфіденційного характеру.<br>3. Необхідний доступ до зовнішніх мереж передачі даних, які мають з'єднання з мережами технічного обслуговування БПЛА.<br>4.Результат взаємодії не визначений, тому що на нього впливає багато ймовірнісних чинників. |
| Нав'язування оператору БПЛА | 1. Порівняно нескладна | 1 Необхідно перебувати в прямій радіовидимості антен апаратури |

| | | |
|---|---|---|
| неправдивої інформації через апаратуру приймання передачі команд і відправки телеметрії | реалізація.<br>2. Можливість спровокувати оператора на дії, необхідні атакуючому.<br>3. Існує ймовірність здійснити збій польотного завдання. | приймання-передавання даних центру управління польотів.<br>2 Немає повного контролю над БПЛА.<br>3. Реалізація впливу і його наслідки сильно залежать від досвіду оператора.<br>4. Необхідне знання використовуваних протоколів зв'язку.<br>5. У разі застосування в каналі СКЗІ (імітовставка або шифрування), вплив стає неможливим. |
| Вплив на приймально-передавальний тракт системи управління БПЛА | 1.Повний контроль над БПЛА з функціоналом оператора. | 1. Необхідне знання використовуваних протоколів зв'язку.<br>2. У разі застосування в каналі СКЗІ (імітовставка або шифрування), вплив стає неможливим.<br>3. Деякі системи зв'язку можуть бути несприйнятливими до цього виду атаки. |
| Вплив на цільове навантаження | 1. Може призвести до зриву польотного завдання. | 1. Не може або дуже рідко може призвести до захоплення управління БПЛА.<br>2 У разі застосування в каналі СКЗІ (імітовставка або шифрування), вплив стає неможливим. |
| Вплив на систему просторового позиціонування БПЛА | 1 Може призвести до часткового або повного захоплення управління.<br>2.Легка реалізація.<br>3.Вплив погано виявляється з боку оператора.<br>4. Для протидії необхідна розробка нових захищених виробів. | 1.Потрібна попередня розвідка апаратури просторового позиціонування.<br>2. Необхідно мати велику кількість різного обладнання. |

Продовження табл. 9.1

| | | |
|---|---|---|
| Вплив на радіоканал | 1.Можливість здійснення повного контролю над БПЛА.<br>2. можливість здійснення спостереження за дією БПЛА без відома оператора.<br>3.Контроль усіх показників телеметрії БПЛА. | 1.Складна реалізація, автоматизація;<br>2. у разі застосування в каналі СКЗІ (імітовставка або шифрування), вплив стає неможливим.<br>3. необхідні знання використовуваних протоколів зв'язку і системи управління БПЛА. |

## 9.3. Розробка моделі загроз на основі виявлених уразливостей БПЛА

Розробка моделі загроз на основі виявлених уразливостей БПЛА - це важлива складова для забезпечення їх безпеки. Для цього потрібно визначити потенційні загрози, які можуть виникнути внаслідок зловживання уразливостями системи управління БПЛА, і розробити заходи з їх запобігання.

Першим кроком у розробці моделі загроз є визначення уразливостей системи управління БПЛА. Уразливість може виникнути внаслідок ряду факторів, таких як: погано спроектована система, помилки в програмному забезпеченні, відкриті порти зв'язку і інше. Після виявлення уразливостей необхідно проаналізувати їх потенційні наслідки і визначити можливі вектори атаки. Очевидно, що КЗ, який використовується для комунікації між БПЛА, є лише бездротовим. Це робить КЗ вразливим до зловмисних атак з боку повітряних зловмисників, що літають поблизу, а також наземних супротивників.

Для розробки моделі загроз можна використовувати різні методики, такі як аналіз ризику, моделювання загроз і інше. Основною метою такої моделі є визначення потенційних загроз, які можуть виникнути внаслідок зловживання уразливостями системи управління БПЛА, і розроблення ефективних заходів з їх запобігання.

### 9.3.1. Підхід, орієнтований на цілі

Дослідження, які проводилися, в основному дотримуємося цільового підходу до моделювання та аналізу загроз безпеці, використовуючи елементи візуальної моделі для явного відображення концепцій, пов'язаних із загрозами [3]. Така модель дозволяє розробникам систем зрозуміти профіль загроз системи шляхом належного вивчення системи, як це зробив би супротивник. Це також допомагає їм визначити кількість і типи ризиків високого рівня безпеки, що загрожують системі [5]. Отримана модель загроз описує потенційні атаки на систему. Одним із застосувань може бути розуміння серйозності атаки та

оцінка рішень, які впливатимуть на безпеку системи протягом тривалого періоду часу [18]. Модель також може бути використана як основа для тестування на проникнення в систему, оскільки система розвивається після декількох ітерацій проектування, розробки та тестування [19].

### 9.3.2. Проста архітектура БПЛА

Базову модель БПЛА можна визначити як комбінацію шести окремих, але взаємозалежних систем: Модуль збору даних, AHRS (система визначення висоти і курсу), NAV (навігаційна система), модуль управління, модуль реєстрації даних і модуль телеметрії [18]. Модуль системи зв'язку при такому підході окремо не показано, оскільки він охоплює всі модулі і через нього проходять всі вхідні/вихідні сигнали управління та передавання даних.

### 9.3.3. Архітектура системи БПЛА

Як показано віще, в систему включено всі КЗ, які є важливими з точки зору безпеки. Компоненти системи покладаються на бездротові КЗ для комунікації один з одним. GCS (наземні станції управління) можуть бути двох типів: локальні та штабні. Штабна станція може бути розташована в центрі управління відповідного відомства/департаменту. Переносні пункти управління - це підклас локальних пунктів управління, які можуть бути КПК (портативними цифровими помічниками), смартфонами, захищеними ноутбуками і інше.

Хоча різні КЗ здаються схожими, існує велика різниця між ними з точки зору безпеки [20]. Зв'язок між супутником і БПЛА - це радіозв'язок LOS (лінія прямої видимості), в той час як зв'язок БПЛА-БПЛА, КПК-БПЛА і локальної ГКС-БПЛА може бути як радіозв'язком LOS, так і зв'язком на основі GPRS/EDGE з використанням існуючої інфраструктури зв'язку. Дослідження показують, що за допомогою БОТ-мереж можна зламати такі мережі за допомогою невеликого БПЛА і бот-майстра шляхом виявлення вразливостей в існуючих мережах Wi-Fi, оскільки методи захисту, що використовуються в мережах Wi-Fi, як відомо, є небезпечними і ненадійними [3].

Загрози для кожного з цих КЗ і компонентів є різними і мають різні вимоги до безпеки. Такі компоненти, як супутник і штабна ГКС, можуть мати певні загрози, але можуть бути не надто вразливими завдяки існуючим заходам безпеки [3, 4].

### 9.3.4. Критерії моделювання безпеки

Зі зростанням використання БПЛА виникає загроза їх використання в злочинних та терористичних актах. У зв'язку з цим, моделювання безпеки БПЛА є надзвичайно важливою задачею, метою якої є забезпечення надійності функціонування цих систем та запобігання непередбачуваним наслідкам.

Критерії моделювання безпеки БПЛА - це сукупність параметрів та метрик, які використовуються для визначення рівня безпеки БПЛА.

Одним з основних критеріїв безпеки БПЛА є забезпечення конфіденційності даних, що обробляються відомствами. Для забезпечення цього критерію використовуються різні методи шифрування та захисту від несанкціонованого доступу до даних.

Оскільки безпека БПЛА включає в себе не тільки захист від зовнішніх загроз, але й можливість безперешкодного використання БПЛА за призначенням, доступність може відноситися до критеріїв моделювання безпеки БПЛА. Доступність означає, наскільки легко для зловмисника отримати доступ до системи БПЛА або її компонентів, що може призвести до викрадення даних або завдання шкоди. Для забезпечення безпеки БПЛА важливо визначити, наскільки легко можливо отримати доступ до системи, та прийняти заходи для зменшення ризиків злому та несанкціонованого доступу.

Іншим важливим критерієм безпеки є забезпечення цілісності даних. Цей критерій вимагає, щоб дані, що передаються через систему БПЛА, не змінювалися під час передачі та не були пошкоджені.

Також серед критеріїв безпеки можна виділити надійність системи управління БПЛА. Для забезпечення цього критерію використовуються різні методи виявлення та виправлення помилок у програмному забезпеченні системи управління.

Більшість наявних моделей безпеки стараються впливати лише на одну властивість системи за один раз, шляхом створення правил політики та забезпечення їх дотримання. Однак у зв'язку з тим, що всі три аспекти (конфіденційність, цілісність та доступність) є дуже чутливими, жоден з них не може бути скомпрометований без наслідків для безпеки системи в цілому. Тому для забезпечення комплексної безпеки системи БПЛА необхідно моделювати всі три аспекти, враховуючи їх взаємодію та вплив одного аспекту на інші.

### 9.3.5 Обмеження

Кілька фірм намагалися визначити різні загрози безпеці та їхній вплив шляхом детального дослідження з використанням значної кількості даних. WhiteHat Security, наприклад, випускає свої звіти з аналізу загроз і вразливостей щороку [21]. Наведений вище аналіз проводився в дуже загальному вигляді для системи БПЛА, і оскільки більшість даних, що стосуються цієї системи, є дуже конфіденційними і чутливими за своєю природою, їх отримання вимагає високого рівня допуску до них. Це пояснює причину відсутності фактичної оцінки загроз або вразливостей на основі даних.

В 2012 році була розроблена модель загроз кібербезпеці БПЛА, вона відображає потенційні вектори загроз безпілотних літальних апаратів, які можна порівняти з існуючими контролюючими заходами для безпілотних літальних апаратів CRMDM [17]. Згідно неї була розроблена модель загроз, яка показана на рисунку 9.1.

9.3.6 Аналіз та моделювання загроз

Аналіз загроз вважається важливим аспектом забезпечення безпеки системи, оскільки він може фактично призвести до виявлення вразливостей в системі [24]. Модель загроз системам БПЛА та можливі шляхи атак, пов'язані з цими загрозами, показано на рисунку 9.1. Розглянемо детально ці загрози, використовуючи модель загроз кібербезпеки.

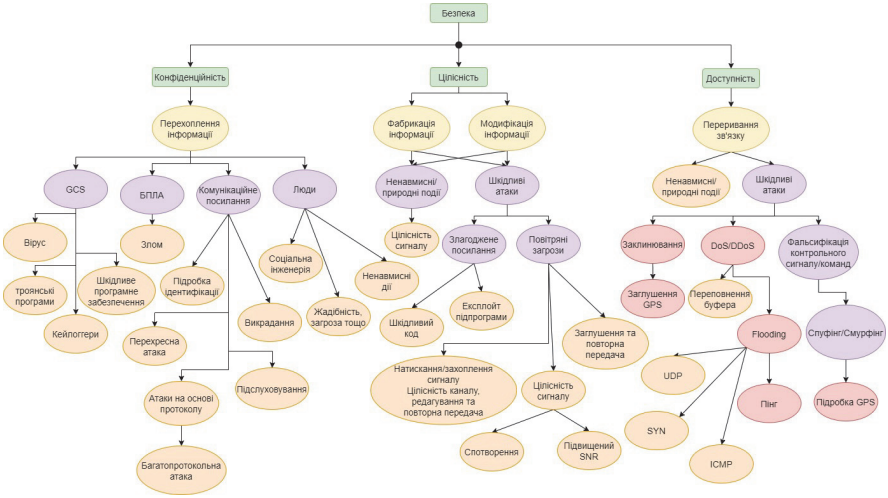



Рисунок 9.1 - Модель загроз кібербезпеки системам БПЛА

***1) Атаки на конфіденційність -***

Ця властивість в першу чергу стосується несанкціонованого доступу до інформації, і найпоширенішим способом порушення безпеки цієї властивості є перехоплення інформації. Чотирма основними компонентами моделі БПЛА, які є вразливими до цього класу атак, є БПЛА, GCS (всіх типів), лінія зв'язку та людина. Загрози для GCS здебільшого мають програмне забезпечення: віруси, шкідливі програми, трояни, кейлоггери тощо. Основною загрозою для БПЛА є хакерство. Слід розуміти, що програмні загрози також можуть впливати на БПЛА, але існує менше способів потрапляння цих загроз на БПЛА. Порушення безпеки GCS або порушення безпеки самого БПЛА може призвести до інших загроз для БПЛА, але потреба в усуненні цих загроз вже виконана, якщо вони вирішуються на рівні GCS.

Основним джерелом порушення безпеки каналів зв'язку між різними компонентами системи є мережеві атаки, такі як злом, підслуховування, підміна ідентифікаційних даних, міжшарові атаки [23] та багатопротокольні атаки [24]. Очевидно, що всі ці атаки можуть бути застосовані не до кожного з наявних в

системі КЗ. Метою об'єднання всіх цих атак в одну групу є виявлення всіх можливих загроз для відповідної лінії зв'язку, а не визначення загроз для кожного типу лінії окремо. При впровадженні належних заходів з пом'якшення наслідків необхідно звернути увагу на те, які атаки насправді впливають на ці канали, і розгорнути відповідні заходи [25]. Що стосується людського фактору, то тенденція до зростання соціальних і ділових мереж спричинила появу нових видів загроз. Деякі з них: соціальна інженерія, "фейкові" онлайн-змагання, шантаж та поведінкова експлуатація.

***2) Атаки на цілісність –***

Цілісність системи може бути порушена за допомогою двох основних операцій: модифікації існуючої інформації та фабрикації нової інформації. Модифікація спрямована на зміну даних під час передачі або зберігання. Природні явища, такі як блискавки, зсуви магнітних полюсів, сонячні спалахи тощо, можуть спричинити певну втрату цілісності та додати небажаного шуму до сигналу. Однак ці природні явища трапляються постійно, і більшість комунікаційних протоколів та обладнання вже мають технології для вирішування цих проблем.

Наступними є штучні загрози, які поділяються на три основні категорії: глушіння, внесення поодиноких або групових помилок та перехоплення/перехоплення каналу/сигналу. Глушіння має на меті перервати зв'язок через перешкоди або зіткнення перед прийомом. Дослідники запропонували багато стратегій захисту від глушіння в бездротових мережах. Для порушення цілісності сигналу найбільш поширеним підходом є спотворення або збільшення SNR (відношення сигнал/шум). Третій спосіб - прослуховування або перехоплення сигналу - є найскладнішим типом атаки, оскільки вимагає великої кількості інформації про частоту сигналу, дальність передачі тощо. Цікаво, що вивченням цих атак займається окрема галузь комунікаційної інженерії під назвою Signal Intelligence [3-10]. Далі йде фабрикація або модифікація інформації, яка включає в себе використання шкідливого коду або існуючих підпрограм системи. Експлуатація підпрограм передбачає атаку на систему шляхом пошуку та використання вразливостей у коді системи, коли противник має достатньо інформації про систему за допомогою спланованої або грубої силової атаки.

***3) Атаки на доступність –***

Основними кібератаками, які можуть вплинути на доступність, є глушіння, фальсифікація сигналів і атаки на відмову в обслуговуванні (DoS) [6, 17, 19]. Як вже зазначалося, передача фальшивих команд або сигналів управління вимагає великої розвідки сигналів. Це може бути серйозною загрозою для доступності системи БПЛА, оскільки хибні сигнали можуть фактично змусити БПЛА приземлитися або атакувати інше місце.

Атаки DoS або DDoS (Distributed DoS) в основному базуються на перевантаженні мережі або переповненні мережевої карти системи, що призводить до того, що система виявляється недоступною. Під час такої атаки система або мережа насправді зайнята обслуговуванням інших "фальшивих"

запитів. Існує три способи запуску такої атаки: flooding, спуфінг/смурфінг і переповнення буфера. Наводнення полягає в тому, що мережа переповнюється одним або декількома типами мережевих пакетів шляхом надсилання декількох пакетів на систему, яку атакують. Зазвичай в такій атаці використовуються SYN, UDP, ICMP і Ping-пакети. Наступним типом атак цього класу є переповнення буфера, метою якого є переповнення буферної пам'яті мережевих карт на пристроях, що використовуються в системі. Smurfing полягає у переповненні системи підробленими широкомовними мережевими пакетами, при цьому цільовій системі здається, що всі пакети надходять з різних адрес.

## 9.4. Оцінка ризиків безпеки та рівня загроз БПЛА

Для оцінки ризиків загрози для БПЛА систем, вони аналізуються з точки зору ймовірності їх виникнення, можливого впливу на окремих користувачів і систему, а також глобального ризику, який вони становлять, дотримуючись стандартної методології оцінки [17]. Оцінка проводиться за трьома критеріями: ймовірність, вплив і серйозність. Показник ймовірності оцінює можливість ініціювання атак. Оцінка наслідків оцінює кінцевий стан системи після атаки. Серйозність ризику розраховується як добуток значень впливу та ймовірності для даної загрози. Підсумовуючи, в таблиці 9.2 показано використане відображення якісної оцінки ризиків у відповідний ранг рівнів ризиків.

У таблиці 9.3 узагальнено результати наведеного аналізу загроз та перелічено відповідні рівні ризиків. Кількісна оцінка рівнів ризику ґрунтується на припущенні, що були впроваджені належні механізми захисту (див. колонку "Пом'якшення безпеки").

Наведений аналіз підтверджує спостереження, що спуфінг і атаки на відмову в обслуговуванні є одними з найбільш серйозних загроз для роїв БПЛА. Також очевидно, що ройова система БПЛА потребує механізму спостереження за загрозами, моніторингу вторгнень і реагування на кібератаки відповідно до ретельно розроблених протоколів безпеки, які активуються при виявленні атаки.

Також видно, що не тільки супротивники, в звичайному розумінні хакери або ображені внутрішні користувачі, є важливими джерелами загроз, але й маніпуляції з навколишнім середовищем також відіграють важливу роль.

***Рекомендації щодо проектування системи безпеки***

На основі аналізу (таблиця 9.2) продемонстровані основні методи зниження рівня безпеки, які необхідно впровадити для протидії загрозам найвищого рівня ризику.

1) *Автентифікація походження повідомлень*: повинен бути впроваджений механізм автентифікації суб'єктів, тобто БПЛА, GCSs і CSs в мережі.

2) *Автентифікація повідомлень та захист цілісності*: Обмін повідомленнями між суб'єктами повинен бути автентифікований і захищений, щоб забезпечити довіру до даних.

3) *Захист конфіденційності*: Шифрування даних необхідне для забезпечення конфіденційності інспекційної місії.

4) *Безпечна реєстрація*: Для зменшення загроз відмов необхідний метод безпечного і захищеного від несанкціонованого втручання ведення журналу дій, включно з контрольними записами аудиту безпеки, щоб зменшити загрози відмов від перевірки.

5) *Довірені обчислення*: Система повинна реалізовувати надійне середовище виконання програмного забезпечення. Бажано впровадити методи виявлення вторгнень для безперервного моніторингу БПЛА.

Під час розробки рішення для перевірки безпеки БПЛА, не менш важливою складовою є рівень довіри, який може бути наданий кінцевим користувачам. Це включає в себе розуміння можливих загроз та способів їх усунення. Окрім того, також важливо враховувати пріоритетність загроз, яка може варіюватися в залежності від конкретної ситуації та умов експлуатації БПЛА. Для забезпечення ефективності та надійності БПЛА, необхідно ретельно проаналізувати потенційні загрози, які можуть виникнути в ході його експлуатації, та розробити механізми захисту та управління ризиками. При цьому необхідно мати глибоке розуміння різноманітних технологій, які використовуються в БПЛА, а також механізмів їх захисту.

Крім того, важливо враховувати весь життєвий цикл БПЛА, включаючи проектування, розробку, випробування та експлуатацію. Необхідно використовувати інноваційні методики, що дозволяють забезпечити максимальний рівень безпеки та надійності, а також захистити БПЛА від можливих загроз. Отже, розуміння можливих загроз та розробка відповідних механізмів захисту є ключем до успішного проектування та реалізації безпечних та надійних БПЛА.

Таблиця 9.2 – Сітка оцінки ризиків

| **Критерії** | **Випадки** | **Обґрунтування** | | **Рейтинг** |
|---|---|---|---|---|
| Ймовірність | Малоймовір-но | Сильно | Низький | 1 |
| | Можливо | Можливо розв'язати | Помірний | 2 |
| | Ймовірний | Неможливо | Високий | 3 |
| Вплив | Низький | Роздратування | Дуже обмежене відключення | 1 |
| | Середній | Втрата сервісу | Обмежене відключення | 2 |
| | Високий | Тривала втрата часу обслуговування | Довгострокова відключення | 3 |
| Тяжкість | Малий | Немає потреби в контрзаходах | | 1-2 |
| | Великий | Загрозу потрібно долати | | 3-4 |
| | Критичний | Високий пріоритет | | 6-9 |

Таблиця 9.3 – Резюме аналізу загроз для системи безпілотних літальних апаратів

| **Опис загрози** | **Зловмисники** | **Об'єкти атаки** | **Пом'якшення наслідків для безпеки** | **Рівні ризику** | | |
|---|---|---|---|---|---|---|
| | | | | Імовірність | Вплив | Тяжкість |
| Spoofing | Шкідливий UAV, GCS, вузол ROS | Підслуховування | Автентифікація походження повідомлень, шифрування даних | 3 | 1 | 3 |
| | Шкідливий UAV, GCS, вузол ROS | Впровадження фальшивих даних | Автентифікація повідомлень, перевірка цілісності | 2 | 3 | 6 |
| | Зловмисний актор у дії | Крадіжка активу, саботаж | Фізичний захист | 3 | 2 | 6 |
| | Зловмисний актор у дії | Введення фальшивих даних GNSS | Автентифікація походження повідомлення, перевірка цілісності | 2 | 3 | 6 |
| Фальсифікація | Зловмисний актор у дії | Крадіжка, захоплення або саботаж | Захист фізичної безпеки, зміцнення системи | 3 | 2 | 6 |
| | UAV, GCS, поганий виробник, погане середовище | Спотворені дані про місію та корисне навантаження | Перевірка цілісності | 2 | 2 | 4 |
| Відмова | Оператор, керівник, поганий виробник | Заперечення конкретних операційних дій, пошкоджених або відсутніх журналів | Журнали з функцією захисту від несанкціонованого доступу та аудит безпеки | 3 | 1 | 3 |
| | Оператор, інженер | Заборона певних конфігурацій та дизайнів | Перевірені та сертифіковані програмні та апаратні компоненти | 3 | 2 | 6 |

Продовження табл. 9.3

| | | | | | | |
|---|---|---|---|---|---|---|
| Розкриття інформації | Зловмисний актор у дії | Поява бездротового зв'язку, підслуховування | Шифрування даних і захист на рівні 2 | 3 | 2 | 6 |
| | Зловмисний актор в Інтернет | Початок IP або ROS-зв'язку, підслуховування | Шифрування даних та рівень 3 захисту | 3 | 2 | 6 |
| | Зловмисник в Інтернеті, поганий хмарний провайдер або шкідливе програмне забезпечення. | Поставити під загрозу хмарне сховище даних | Захист периметра (брандмауер) | 3 | 2 | 6 |
| Відмова в обслуговуванні | Шкідливий UAV, GCS, вузол ROS | Впровадження хибного сигналу | Перевірка автентичності повідомлення | 2 | 3 | 6 |
| | Зловмисний актор у дії | Глушіння радіосигналу | Протокол безпеки | 3 | 2 | 6 |
| Відмова в обслуговуванні | Зловмисний актор у дії | Глушіння сигналу GNSS/GPS | Протокол безпеки, викликати вторинну навігаційну систему | 3 | 2 | 6 |
| | Шкідливий інтернет-вузол | Розподілена DoS-атака через Інтернет | Зміцнення системи, захист периметра, моніторинг мережі, протокол безпеки. | 3 | 1 | 3 |
| Відмова в обслуговуванні | Поганий виробник, погане середовище. | Відмова від зарядки, позбавлення енергії, відключення/вимкнення живлення. | Протокол безпеки | 2 | 3 | 6 |
| Підвищення привілеїв | Зловмисний актор, поганий виробник. | Зараження шкідливим програмним забезпеченням, витік даних, порушення роботи та створення "чорних ходів" | Захист від шкідливих програм, виявлення вторгнень, захист периметра | 3 | 2 | 6 |

Продовження табл. 9.3

| | | | | | | |
|---|---|---|---|---|---|---|
| Підвищення привілеїв | Зловмисник або шкідливе програмне забезпечення. | Використовувати інші види кібератак | Впровадження принципу найменших привілеїв, аудит аудиту безпеки, зміцнення системи | 3 | 2 | 6 |

Рівні ризику класифікуються як малоймовірні (1), можливі (2), ймовірні (3) для ймовірності та низькі (1), середні (2) і високі (3) для впливу, див. табл. 9.2.

## 9.5 Висновок

У висновку можна зазначити, що використання БПЛА стає дедалі поширенішим, однак це також призводить до збільшення вразливості системи управління БПЛА та ризику для безпеки як БПЛА, так і навколишнього середовища та людей. Тому необхідно розробляти та впроваджувати моделі безпеки для БПЛА, які враховують різні критерії безпеки, включаючи конфіденційність, цілісність та доступність системи управління.

На основі проведеного аналізу було виявлено основні вразливості системи управління БПЛА та розроблено модель загроз. Це дає змогу оцінити ризики та вжити заходів для підвищення безпеки використання БПЛА.

Для того, щоб гарантувати безпеку під час використання БПЛА, необхідно дотримуватися відповідних стандартів і правил, як-от "Правила ведення робіт з використанням безпілотних літальних апаратів" та "Авіаційні правила України. Частина XXIII. Безпілотні літальні апарати". Також слід враховувати міжнародні стандарти цивільної авіації та рекомендації Європейського агентства авіаційної безпеки.

Однак для забезпечення цілковитої безпеки необхідно постійно вдосконалювати системи управління БПЛА та розробляти нові методи захисту від загроз. Важливим кроком у цьому напрямі може стати використання штучного інтелекту і машинного навчання для виявлення і запобігання загрозам.

Таким чином, подальший розвиток і використання БПЛА вимагає посиленої уваги до питань безпеки та захисту від загроз. Впровадження відповідних заходів і технологій дасть змогу не тільки забезпечити безпеку, а й розкрити нові можливості для застосування БПЛА в різних сферах.

## Література

1. Chung T.H., Jones K.D., Day M.A., Jones M., Clement M. 50 vs. 50 by 2015: Swarm vs. Swarm UAV live-fly competition at the naval postgraduate school. AUVSI, 2013, pp. 1792– 1811.

2. Zikratov I.A., Kozlova E.V., Zikratova T.V. Vulnerability analysis of robotic systems with swarm intelligence. Scientific and Technical Journal of Information Technologies, Mechanics and Optics, 2013, no. 5, pp. 149–154.

3. Tutubalin P.I., Kirpichnikov A.P. Ensuring information security of functioning of unmanned reconnaissance complexes. Vestnik KSTU, 2017, vol. 20, no. 21, pp. 86–92.

4. Higgins F., Tomlinson A., Martin K.M. Threats to the swarm: Security considerations for swarm robotics. International Journal on Advances in Security, 2009, vol. 2, no. 2&3, pp. 288–297.

5. Sedjelmaci H., Senouci S.M. Cyber security methods for aerial vehicle networks: taxonomy, challenges and solution. Journal on Advances in Security. 2009. V. 2. N 2&3. P. 288– 297.

6. Sedjelmaci H., Senouci S.M. Cyber security methods for aerial vehicle networks: taxonomy, challenges and solution // The Journal of Supercomputing. 2018. P. 1–17. doi: 10.1007/s11227-018-2287-8.

7. CNN Wire Staff. Obama says U.S. has asked Iran to return drone aircraft. 2011. URL: http://edition.cnn.com/2011/12/12/world/meast/iran-us-drone (accessed 15.04.2021).

8. Noah Shachtman. Exclusive: Virus Hits U.S. Drone Fleet. URL: http://www.wired.com/dangerroom/2011/10/virus-hits-drone-fleet (accessed 15.04.2021).

9. Lolita C. Baldor. Flashy drone strikes raise status of remote pilots. The Boston Globe. 2012. URL: http://www.bostonglobe.com/news/nation/2012/08/11/air-force-works-fillneed-for-drone-pilots/ScoF70NqiiOnv3bD3smSXI/story.html (accessed 15.04.2021).

10. Patricia K. Freeman, Robert S. Freeland. Agricultural UAVs in the U.S.: potential, policy, and hype // Remote Sensing Applications: Society and Environment, Volume 2, December 2015, Pages 35-43. doi: 10.1016/j.rsase.2015. 10.002.

11. "Rules for conducting work using unmanned aerial vehicles" (Order of the State Service of Ukraine for Ecology and Natural Resources No. 616 of 21.07.2015) URL: https://zakon.rada.gov.ua/laws/show/z1119-15#n5 (accessed 19.08.2022).

12. "Aviation rules of Ukraine. Part XXIII. Unmanned Aerial Vehicles" (DSTU-3745:2018) URL: https://zakon.rada.gov.ua/laws /show/z1143-18#n10 (accessed 19.08.2022).

13. "Pilotless Aerodynamic Systems. Technical requirements" (DSTU EN 16605:2015) URL: https://zakon.rada.gov.ua/laws/show/z1207-15#n4 (accessed 19.08.2022).

14. International Civil Aviation Standards (ICAO) URL: https://www.icao.int/ (accessed 19.08.2022). 26.

15. European Aviation Safety Agency (EASA) URL: https://www.easa.europa.eu/ (accessed 19.08.2022).

16. STANAG 4586 (NATO Standardization Agreement 4586) is a NATO Standard Interface of the Unmanned Control System (UCS) Unmanned Aerial Vehicle (UAV). URL: https://defense-update.com/products/s/stanag_4586.htm (accessed 19.08.2021).

17. Y. Javaid, W. Sun, V. K. Devabhaktuni, and M. Alam, "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," in 2012 IEEE Int. Conf. Technol. Homel. Secur., November 2012, pp. 585–590. [Online]. Available: https://doi.org/10.1109/THS.2012.6459914

18. F. Swiderski and W. Snyder, "Threat Modeling", Microsoft Press, 2004.

19. J. Evans, G. Inalhan, Jang, Jung S, R. Teo and C. J. Tomlin, "Dragon Fly: a versatile UAV platform for the advancement of aircraft navigation and control", 20th DASC Conference on Digital Avionics System, 2001.

20. D. Rudniskas, Z. Goraj and J. Stankunas, "Security Analysis of UAV Radio Communication System", Taylor & Francis International Research Journal on Aviation, Vol. 13, Issue 4, Pages 116-121, 2009.

21. Jeremiah Grossman, "Top Website Vulnerabilities: Trends, Business Effects & How to Fight Them", March 03 2011. https://www.white hatsec.com/assets/presentations/11PPT/PPT_topwebvulns_030311.pdf (accessed 15.11.2022).

22. R. Crook, D. Ince, L. Lin, and B. Nuseibeh, "Security requirements engineering: When anti-requirements hit the fan", Proceedings of International IEEE Requirements Engineering Conference, RE-2002

23. W. Wang, Y. Sun, H. Li and Z. Han, "Cross-Layer Attack and Defense in Cognitive Radio Networks", IEEE Globe Communication Conference (Globecom), Miami, FL, Nov.-Dec. 2010.

24. Jim Alves-Foss, "Multi-Protocol Attacks and the Public Key Infrastructure", 21st National Information Systems Security Conference, Arlington, Virgina, USA, Sept 1998.

25. C. M. Schneidera, A. A. Moreirab, J. S. Andrade, S. Havlinc and H. J. Herrmanna, "Mitigation of malicious attacks on networks", Proceedings of the National Academy of Sciences of the USA, Vol. 108, No. 10.

# ЧАСТИНА ІІІ. МЕТОДИ І ТЕХНОЛОГІЇ ПОБУДОВИ БЕЗПЕЧНИХ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

## 10. ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ МОНІТОРИНГУ ПОТЕНЦІЙНО НЕБЕЗПЕЧНИХ ОБЄКТІВ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ЛІТАЮЧИХ КРАЙОВИХ ОБЧИСЛЕНЬ

**С. В. Скоробогатько, Г. В. Фесенко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

### 10.1. Вступ

Літаючі граничні обчислення (ЛГО) на основі БПЛА наразі розглядаються у якості життєво важливої технології для впровадження багатьох методів для IoT-застосунків перспективних систем моніторингу (СМ) потенційно небезпечних об'єктів (ПНО) наступного покоління. Безумовно, для того, щоб архітектура СМ ПНО було більш гнучкою, ресурсно- та енергоефективною, ця технологія може реалізовуватися у поєднанні з технологіями літаючих хмарних (ЛХО) та туманних (ЛТО) обчислень, а також наземних граничних (НГО), хмарних (НХО) та туманних (НТО) обчислень. При цьому можливості цих технологій можуть бути значно розширені за рахунок використання ними для вирішення певних завдань методів штучного інтелекту (ШІ). Завдяки своїй універсальності та простоті розгортання, БПЛА у такій архітектурі можуть відігравати різні ролі:

- виступати у ролі мобільних пристроїв, які перевантажують свої обчислення на наземний сервер;
- діяти як багатоцільова підсистема ЛГО (ЛХО, ЛТО), що відповідає за моніторинг групи мобільних кінцевих вузлів та може одночасно служити ретранслятором або шлюзом між мобільними кінцевими вузлами та наземним сервером. Тому, актуальними є питання дослідження особливостей ЛГО, ЛХО та ЛТО в контексті їх застосування в інтересах СМ ПНО.

### 10.2. Порівняльний аналіз технологій літаючих хмарних, граничних і туманних обчислень

У [1] ЛХО, ЛГО та ЛТО розглядаються як складові технології Інтернету літаючих речей (Internet of Flying Things (IoFT)), відомої також під назвою «Інтернет дронів» (Internet of drones (IoD)). IoFT (IoD) – це багатошарова архітектура, яка акумулює переваги бездротових динамічних (самоорганізованих) літаючих мереж (Flying Ad hoc Networks (FANET)) та Інтернету речей (Internet of Things (IoT)) і призначена для управління літаючою

мережею і надання швидкого доступу БПЛА до контрольованого простору, інтернет-ресурсів і хмарного середовища.

Автори статті [2] запропонували ресурсно-орієнтовану архітектуру, яка призначена для полегшення моделювання ресурсів і послуг, що надаються БПЛА. БПЛА при цьому оснащені платою Arduino, бортовим Wi-Fi обладнанням та виступають у ролі серверів, доступ до хмарних ресурсів яких можна отримати за допомогою інтерфейсів прикладного програмування. У [3] автори розширили можливості свого прототипу, представленого у [2] за рахунок інтеграції плати Arduino з датчиками вимірювання вологості й температури повітря, а також створення можливості керування цими датчиками через інтерфейс за допомогою веб-служб RESTful. Автори [4] представили хмарну архітектуру, яка призначена для забезпечення ефективної взаємодії БПЛА та бездротових сенсорних мереж. У цій архітектурі рівень фізичних ресурсів БПЛА є відокремленим від рівня керування і включає програмне забезпечення, програмно визначені мережі й мережеву функціональну віртуалізацію.

Робота [12] була присвячена описанню особливостей функціонування архітектури ЛГО, де БПЛА забезпечують надання необхідних послуг користувачам у зонах стихійних лих з пошкодженою наземною інфраструктурою зв'язку. Також у цій роботі надано рекомендації щодо оптимізації кількості й місць розміщення БПЛА для більш ефективної реалізації граничних обчислень в інтересах користувачів. У роботі [13] запропоновано архітектуру для наземно-повітряної інтегрованої мобільної периферійної мережі з назвою AG630 MEN, у складі якої розгортаються БПЛА, які відіграють роль граничних мережевих контролерів для ефективного розподілу обчислювальних ресурсів і ресурсів зберігання даних. Автори статті [14] демонструють можливості розробленого ними фреймворку, який поєднує в собі можливості наземних транспортних засобів і БПЛА щодо розгортання граничних серверів для організації зв'язку, проведення необхідних обчислень і забезпечення зберігання необхідної інформації. Результати проведених авторами цієї статті експериментів підтвердили, що застосування розробленого фреймворку забезпечує високу мобільність і пропускну здатність, а також низьку затримку. Для гарантування високої якості обслуговування для ресурсомістких та он-лайн застосунків у статті [15] запропонована гібридна модель хмарних і граничних обчислень для роїв БПЛА. Ця модель розширює ємність ресурсів БПЛА за рахунок використання граничних серверів, які здатні обробляти дані з низькою затримкою. Також ця стаття описує алгоритм взаємодії граничних та хмарних обчислень для оброблення та зберігання великих даних у хмарі.

Представлені у [23] дослідження стосувалися проблеми застосування ЛТО в інтересах Індустрії 4.0. У цій статті була детально описана структура, у якій БПЛА, що реалізують ЛТО, розвантажують завдання, виконувані наземними датчиками, а також за допомогою жадібного алгоритму оптимізують розподіл таких завдань з метою максимізації їх виконаної

кількості за визначений проміжок часу. Представлена авторами [24] система ЛТО під назвою UAVFog, використовуючи туманні обчислення та мобільність БПЛА, дозволяє забезпечити зберігання необхідної кількості даних, гнучкий зв'язок, низьку затримку для IoT-застосунків, а також постачає послуги Інтернету речей: брокерські послуги й послуги на основі місцезнаходження.

На основі розглянутих основних публікацій за цією тематикою авторами було проведено порівняльний аналіз технологій ЛХО, ЛГО та ЛТО з визначенням особливостей архітектури рішень на їх основі, а також основних переваг та недоліків. Результати аналізу подані у таблиці 1.

Як ми можемо бачити з таблиці 10.1, архітектури ЛГО та ЛТО мають переваги над архітектурами ЛХО у гнучкості та енергоефективності, однак поступаються їм у потужності засобів оброблення та зберігання інформації.

Таблиця 10.1 – Порівняльний аналіз технологій літаючих хмарних, граничних та туманних обчислень

| Вид літаючих обчислень | Особливості архітектури | Основні переваги | Основні недоліки |
|---|---|---|---|
| Літаючі хмарні обчислення | Централізована обробка. Швидкий доступ через Інтернет до великої кількості даних. | Масштабованість. Економічна ефективність. Використання надійного TCP/IP протоколу. | Великий час затримки. Обмежена пропускна здатність. Вразливості системи безпеки. Відсутність автономного режиму. Проблема обробки інформації у разі, якщо багато пристроїв надсилають дані одночасно. Обмежений ресурс батареї. Єдина точка відмови. |
| Літаючі граничні обчислення | Немає потреби у стаціонарній комунікаційній інфраструктурі. Літаючий вузол діє як підсистема комунікації та зв'язку. Літаючий вузол знаходиться ближче до кінцевих пристроїв. | Гнучкість. Масштабованість. Енергоефективність. Здатність працювати з мобільними кінцевими пристроями. Можливість автономного виконання процесів, правил та алгоритмів. | Обмежений ресурс батареї літаючого вузла. Єдина точка відмови. |

Продовження табл. 10.1

| | | | |
|---|---|---|---|
| Літаючі туманні обчислення | Децентралізована обробка. Поширює можливості хмарного середовища до границі мережі. | Гнучкість. Масштабованість. Енергоефективність. Низька затримка передачі даних та кращий взаємозв'язок з кінцевими пристроями. Поліпшені можливості використання технологій бездротового доступу. Розширені можливості для застосунків, що працюють у реальному часі. | Обмежений ресурс батареї літаючого вузла. Єдина точка відмови. Дані можуть надсилатися до літаючого вузла складними маршрутами, що збільшує імовірність їхньої часткової або повної втрати. |

10.3. Варіанти схем організації літаючих хмарних, граничних і туманних обчислень

Враховуючи підходи щодо організації ЛХО, ЛГО та ЛТО, розглянуті у проаналізованих джерелах і варіанти архітектур з реалізацією таких обчислень, поданих у [28], розглянемо приклади різних варіантів схем організації ЛХО, ЛГО та ЛТО (рис. 10.1-10.3). Запропонований на рис. 10.1 варіант схеми організації ЛХО може бути застосований у сценарії підтримки прийняття рішень під час ліквідації надзвичайних ситуацій, коли всі кінцеві пристрої (КП) КП-1, КП-2, … КП-m можуть обмінюватися інформацією один з одним, але не мають доступу до зовнішнього світу (немає Інтернету). Локальні послуги (наприклад, розвантаження завдань) надаються на рівні наземних граничних обчислень (шляхом застосування наземних вузлів граничних обчислень НВГО-1, НВГО-2,… НВГО-k) або на рівні наземних туманних граничних обчислень (шляхом застосування наземних вузлів туманних обчислень НВТО-1, НВТО-2, … НВТО-r). Глобальні ж послуги (наприклад, збирання даних, забезпечення безпекових функцій, застосування обчислювальних ресурсів, підтримка прийняття рішень) надаються флотом БПЛА, який виступає у ролі підсистеми літаючих хмарних обчислень (ПсЛХО), а БПЛА флоту – у ролі літаючих вузлів хмарних обчислень ЛВХО-1, ЛВХО-2, … ЛВХО-s.

У варіанті схеми ЛГО (рис. 10.2) флот БПЛА, який виступає у якості підсистеми літаючих граничних обчислень (ПсЛГО), перебуває поблизу джерел даних (кінцевих пристроїв КП-1, КП-2, … КП-m), надає їм необхідні послуги та здійснює необхідні для них обчислення на літаючих вузлах граничних обчислень ЛВГО-1, ЛВГО-2, … ЛВХО-n. Близькість зазначених вузлів до джерел даних дозволяє зменшити час затримки, покращити пропускну здатність, а також збільшити термін служби мережі за рахунок більш ефективного використання ресурсу батареї кінцевих пристроїв.

Якщо ж ПсЛГО не може надати необхідну послугу, вона створює можливість надання цієї послуги за допомогою НХО.

У варіанті схеми ЛТО (рис. 10.3) флот БПЛА, який виступає у ролі підсистеми літаючих туманних обчислень (ПсЛТО), а БПЛА флоту – у ролі літаючих вузлів туманних обчислень ЛВТО-1, ЛВТО-2, … ЛВТО-p, поєднує за допомогою бездротових каналів наземні хмарні сервери та кінцеві пристрої КП-1, КП-2, … КП-m з метою забезпечення більш високої ємності зберігання, швидкості обчислень, а також невеликого часу затримки для кінцевих пристроїв КП-1, КП-2, … КП-m.

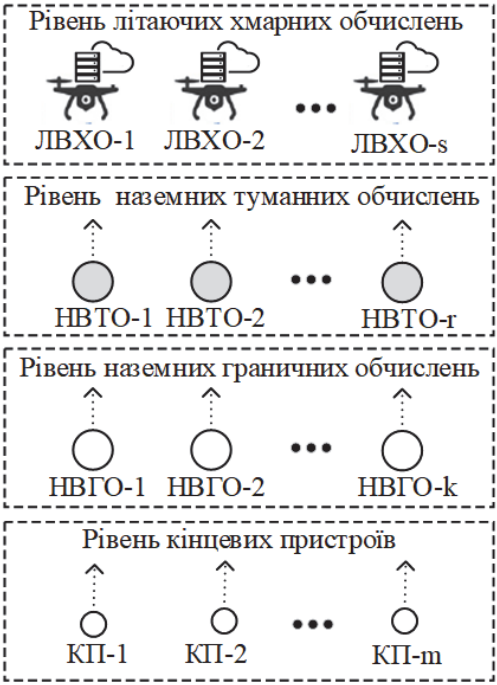



Рисунок 10.1 – Варіант схеми організації літаючих хмарних обчислень

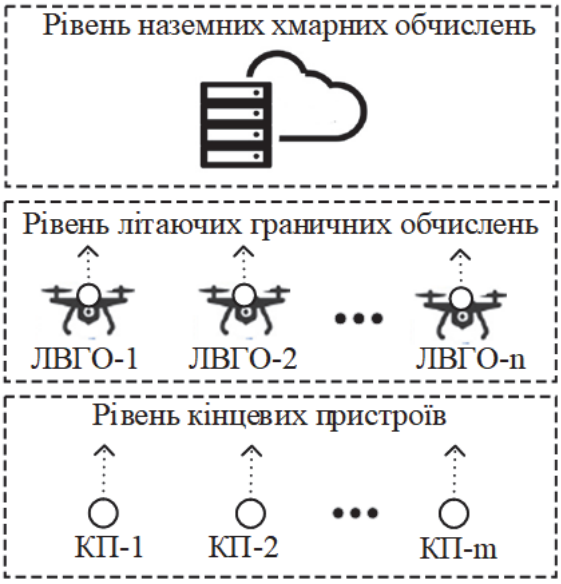



Рисунок 10.2 – Варіант схеми організації літаючих граничних обчислень

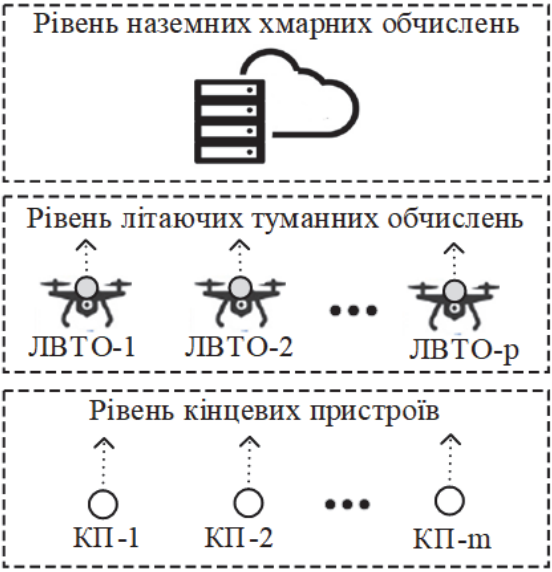



Рисунок 10.3 – Варіант схеми організації літаючих туманних обчислень

10.4. Компоненти перспективної системи моніторингу потенційно небезпечних об’єктів

На думку авторів, ЛХО, ЛГО та ЛТО разом з наземними видами таких обчислень можуть бути затребувані під час розробки перспективних СМ ПНО. Розглянемо перспективну СМ ПНО, до якої входять такі основні компоненти:

- ПНО, який містить об’єкти контролю, до яких, як правило, належать критичні з точки зору безпеки технологічні установки;
- кризовий центр (КЦ), призначений для відпрацювання рішень, спрямованих на попередження та ліквідацію наслідків аварій на ПНО, а також прогнозування виникнення таких аварій і оцінки їх наслідків;
- флот БПЛА, який здійснює функції збору, часткової обробки та передачі моніторингової інформації до КЦ;
- пункт дистанційного пілотування (ПДП), який забезпечує здійснення зовнішнім пілотом (оператором) керування та контроль БПЛА на землі та в повітрі;
- група зовнішніх експертів (ГЗЕ), які дистанційно беруть участь разом з відповідним персоналом КЦ у відпрацювання рішень, спрямованих на попередження та ліквідацію наслідків аварій на ПНО.

Зважаючи на виконувані СМ ПНО завдання, автори пропонують використовувати компонентами СМ ПНО ті чи інші види літаючих та наземних обчислень відповідно до таблиці 10.2. Для реалізації цих видів обчислень у складі компонентів СМ ПНО можуть розгортатися одна або декілька підсистем (таблиця 10.3).

Таблиця 10.2 – Варіант застосування хмарних, граничних та туманних обчислень компонентами СМ ПНО

| Види обчислень | | Компоненти СМ ПНО | | | | |
|---|---|---|---|---|---|---|
| | | ПНО | Флот БПЛА | ПДП | КЦ | ГЗЕ |
| Хмарні обчислення | літаючі | – | + | – | – | – |
| | наземні | – | – | + | + | + |
| Граничні обчислення | літаючі | – | + | – | – | – |
| | наземні | + | – | – | – | – |
| Туманні обчислення | літаючі | – | + | – | – | – |
| | наземні | – | – | – | – | – |

З таблиці 10.2 видно, що найбільш затребуваними є хмарні обчислення, які реалізуються як літаючим компонентом (флотом БПЛА), так і трьома наземними компонентами (ПДП, КЦ, ГЗЕ), а найменш затребуваними – туманні обчислення, які реалізуються тільки літаючим компонентом (флотом БПЛА).

Таблиця 10.3 показує, що найбільша кількість підсистем може бути розгорнута у складі флоту БПЛА, оскільки на нього можуть бути покладені завдання з проведення одразу трьох видів обчислень – ЛХО, ЛГО та ЛТО.

Таблиця 10.3 – Підсистеми у складі компонентів СМ ПНО, створювані для реалізації хмарних, граничних та туманних обчислень

| Компонент СМ ПНО | Назва підсистеми |
|---|---|
| ПНО | ПсНГО-ПНО |
| Флот БПЛА | ПсЛХО-Ф |
| | ПсЛГО-Ф |
| | ПсЛТО-Ф |
| ПДП | ПсНХО-ПДП |
| КЦ | ПсНХО-КЦ |
| ГЗЕ | ПсНХО-ГЗЕ |

## 10.5. Перспективи використання методів штучного інтелекту в системах моніторингу потенційно небезпечних об'єктів

Застосування ШІ для мережевих завдань набуло популярності протягом останніх кількох десятиліть. Наприклад, ШІ широко використовується в мережевій сфері завдяки своїй здатності взаємодіяти зі складним середовищем для інтелектуалізації процесу прийняття рішень. Використання методів ШІ може покращувати продуктивність мережі в багатьох субдоменах, таких як розподіл ресурсів, прогнозування й класифікація мережевого трафіку, контроль перевантаження та маршрутизація. Підсистеми СМ ПНО, які формують як наземні, так і літаючі мережі на основі БПЛА, а також використовують ЛХО, ЛГО та ЛТО для розширення своїх можливостей, повинні забезпечувати безперебійне з'єднання, відповідати вимогам якості обслуговування для багатьох кінцевих пристроїв, обробляти величезний обсяг даних, створених фізичним середовищем.

Методи ШІ, які пропонують надійний аналіз, навчання, оптимізацію та можливості інтелектуального розпізнавання, можуть бути інтегровані в підсистеми СМ ПНО для інтелектуальної оптимізації продуктивності, виявлення необхідної моніторингової інформації, розширеного навчання, організації структури та підтримки прийняття складних рішень щодо прогнозування наслідків аварій на ПНО та реагування на такі аварії.

На підставі проведеного аналізу авторами було сформовано перелік завдань, виконуваних підсистемами СМ ПНО з використанням різних методів ШІ (таблиця 4). Методами, що використовуються, є наступні [29]:

- DL – deep learning (глибоке навчання);
- DSL – deep supervised learning (глибоке кероване навчання);

- DRL – deep reinforcement learning (глибоке навчання з підкріпленням);
- FI – fuzzy inference (нечітке виведення);
- FL – federated learning (федеративне навчання);
- GA – genetic algorithm (генетичний алгоритм);
- RL – reinforcement learning (навчання з підкріпленням);
- RL-ACO – reinforcement learning based on ant-colony optimization (навчання з підкріпленням на основі алгоритму оптимізації мурашиної колонії).

Як ми можемо бачити з таблиці 10.4:

- найбільш затребуваним є метод RL, який використовується для вирішення всього спектру представлених завдань та в інтересах всіх підсистем СМ ПНО;
- найменш затребуваним є метод RL-ACO, який використовується тільки для вирішення завдань розподілу ресурсів в інтересах п'яти із семи підсистем СМ ПНО;
- найбільше методів штучного інтелекту (RL, DRL, GA, DL, FI, FL) використовується під час вирішення завдань, пов'язаних з підтримкою прийняття рішень;
- найменше методів (RL) використовується для вирішення завдань, пов'язаних з плануванням маршрутів руху БПЛА.
- Далі автори проводять огляд використання ШІ в СМ ПНО представленого у роботі [30]. Запропонована реалізація демонструє використання ШІ та ЛГО для дистанційного зондування. Граничні обчислення з використанням ШІ для БПЛА можуть забезпечити низку переваг, а саме:
- мала затримка;
- підвищена ефективність;
- підвищена надійність;
- покращення конфіденційності.

Таблиця 10.4 – Завдання, виконувані підсистемами СМ ПНО з використанням різних методів штучного інтелекту

| Завдання | Метод ШІ | Підсистеми СМ ПНО | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | ПсЛХО-Ф | ПсЛГО-Ф | ПсЛТО-Ф | ПсНГО-ПНО | ПсНХО-ПДП | ПсНХО-КЦ | ПсНХО-ГЗЕ |
| Розвантаження обчислень | RL | + | + | + | + | + | + | + |
| | DRL | + | + | + | + | + | + | + |
| | GA | + | + | + | + | + | + | + |
| | DL | + | + | + | + | + | + | + |
| | FI | + | + | + | + | + | + | + |

Продовження табл. 10.4

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Розподіл ресурсів | RL | + | – | – | + | + | + | + |
| | DRL | + | – | – | + | + | + | + |
| | GA | + | – | – | + | + | + | + |
| | RL-ACO | + | – | – | + | + | + | + |
| Підтримка прийняття рішень | RL | + | – | – | – | – | + | + |
| | DRL | + | – | – | – | – | + | + |
| | GA | + | – | – | – | – | + | + |
| | DL | + | – | – | – | – | + | + |
| | FI | + | – | – | – | – | + | + |
| | FL | + | – | – | – | – | + | + |
| Забезпечення безпеки | RL | + | + | + | + | + | + | + |
| | DRL | + | + | + | + | + | + | + |
| | GA | + | + | + | + | + | + | + |
| | DL | + | + | + | + | + | + | + |
| | FL | + | + | + | + | + | + | + |
| Планування маршрутів руху БПЛА | RL | + | – | + | – | + | – | – |

Архітектура, представлена у роботі [30], складається з чотирьох основних рівнів: рівень БПЛА, рівень флоту БПЛА, хмарний рівень, користувацький рівень. Далі розглянемо докладніше ці чотири рівні.

Рівень БПЛА оснащений засобами обробки та зберігання даних для виконання завдань ШІ. Граничні обчислення, у даній архітектурі, використовуються для обробки вихідних даних. БПЛА оснащені камерами для фіксування об'єктів та графічним процесором для обробки зображень. Для зв'язку та передачі даних з БПЛА використовуються мережеві інтерфейси (наприклад 4G/5G або WiFi).

Флот БПЛА складається з кластера БПЛА, оснащених датчиками камер і пристроями зі штучним інтелектом, які координують свої дії для виконання спільної місії, наприклад, відстеження змін на території електростанції.

Хмарний рівень. Хмара відповідає за зберігання, маніпулювання та візуалізацію даних. Основні обчислювальні задачі виконуються на рівні БПЛА, тому хмарна система не потребує великих/сучасних обчислювальних ресурсів, що значно знижує вартість розгортання, оскільки хмарні системи на базі графічних процесорів, як правило, дорожчі, ніж системи які використовують звичайні цифрові професори (ЦП).

Користувацький рівень взаємодіє з хмарним рівнем через площину користувача, яка забезпечує доступ до авторизованих хмарних ресурсів і дозволяє їм взаємодіяти, контролювати і керувати БПЛА для роботи. Кінцеві користувачі використовують інтерактивні інформаційні панелі для моніторингу

стану БПЛА у режимі реального часу, надсилання команд та отримання інформації у реальному часі, яка була оброблена застосунками з використанням алгоритмів DL, розташованими на борту БПЛА або у хмарі. Користувачі можуть отримати доступ до системи за допомогою API веб-сервісів.

Модель ШІ, яка використовується у даній реалізації, складається з трьох модулів, а саме: модуля виявлення, модуля прискорення моделі та модуля відстеження.

Модуль виявлення заснований на алгоритмі YOLOv7 [31]. Він встановив сучасний рівень як за точністю, так і за швидкістю роботи. Для відстеження об'єктів в реальному часі використовується трекєр DeepSORT [32], який є ефективним алгоритмом, що використовується для відстеження об'єктів у реальному час. Система виявлення та відстеження об'єктів обробляє кожен новий кадр, спочатку застосовуючи YOLOv7 до всього кадру, щоб отримати граничні значення та оцінки достовірності для всіх виявлених об'єктів. Потім ці граничні значення вводяться в DeepSort, який створює відповідні пари виявлених об'єктів, а також списки невідповідних пар виявлених об'єктів. Для кожної створеної пари об'єктів система перевіряє, чи слід її відкинути, обробити далі або надіслати на сервер.

Для прискорення обробки даних у запропонованій схемі використовується окремий модуль. Цей модуль використовує як основу існуючий фреймворк TensorRT [33], який дозволяє оптимізувати моделі глибокого навчання (DL). TensorRT може оптимізувати моделі, зменшуючи точність параметрів моделі та мінімізуючи пам'ять, необхідну для їх зберігання, що дозволяє моделі ефективніше працювати на граничних пристроях з обмеженими ресурсами. TensorRT також оптимізує моделі шляхом об'єднання слоїв, фреймворк об'єднує кілька слоїв нейронної мережі в один для прискорення виведення моделі. Це особливо важливо для систем, які вимагають обробки в реальному часі на граничних пристроях, де затримка є критичною.

Інформація про місцезнаходження об'єктів отримується за допомогою фотограмметрії та метаданих. Використання метаданих, таких як висота над рівнем моря і місцезнаходження БПЛА за GPS, розмір зображення і відкалібрована фокусна відстань, забезпечує надійну основу для визначення місцезнаходження об'єктів.

Ще одним важливим аспектом, який необхідно враховувати при розробці безпілотних систем з можливостями бортового ШІ, є безпека. Зв'язок безпілотників є вразливим до кібератак, що робить вкрай важливим захист даних, які передаються між БПЛА та хмарою. Впровадження заходів безпеки, таких як шифрування та протоколи автентифікації, може захистити систему від несанкціонованого доступу та витоку даних. Крім того, реалізація заходів фізичної безпеки, таких як захист бортового обладнання ШІ, може запобігти втручанню зловмисників у роботу системи. Ці заходи безпеки необхідно впроваджувати на кожному етапі розробки і розгортання системи, щоб забезпечити безпеку і конфіденційність даних, зібраних БПЛА. Тим не менш, ці

заходи можуть вплинути на швидкість роботи системи таким чином, що їх ще належить дослідити.

## 10.6 Висновки

Представлено результати порівняльного аналізу технологій ЛХО, ЛГО та ЛТО з визначенням особливостей архітектури рішень на їх основі, а також основних переваг та недоліків. Зазначені результати показали, що архітектури ЛГО та ЛТО мають переваги над архітектурами ЛХО у гнучкості та енергоефективності, однак поступаються їм у потужності засобів оброблення та зберігання інформації.

Розглянуто варіанти багаторівневих схем організації ЛХО, ЛГО та ЛТО й показані особливості взаємодії літаючих вузлів з кінцевими пристроями.

Запропоновано варіант застосування ЛХО, ЛГО та ЛТО компонентами СМ ПНО. У цьому варіанті найбільш затребуваними є хмарні обчислення, які реалізуються як літаючим компонентом (флотом БПЛА), так і трьома наземними компонентами (ПДП, КЦ, ГЗЕ), а найменш затребуваними – туманні обчислення, які реалізуються тільки літаючим компонентом (флотом БПЛА). Показано, які підсистеми у складі компонентів СМ ПНО повинні бути створені для реалізації вказаних обчислень. Найбільшу кількість підсистем необхідно розгорнути у складі флоту БПЛА, оскільки на нього можуть бути покладені завдання з реалізації одразу трьох видів обчислень – ЛХО, ЛГО та ЛТО.

Запропоновано варіант використання методів ШІ для розширення можливостей ЛХО, ЛГО та ЛТО під час виконання СМ ПНО завдань з розвантаження обчислень, розподілу ресурсів, підтримки прийняття рішень, забезпечення безпеки й планування маршрутів руху БПЛА. Відповідно до цього варіанту найбільш затребуваним є метод RL, який використовується для вирішення всього спектру представлених завдань та в інтересах всіх підсистем СМ ПНО, а найменш затребуваним – метод RL-ACO, який використовується тільки для вирішення завдань розподілу ресурсів в інтересах п’яти із семи підсистем СМ ПНО. Найбільше методів штучного інтелекту (RL, DRL, GA, DL, FI, FL) використовується під час вирішення завдань, пов’язаних з підтримкою прийняття рішень, а найменше – під час вирішення завдань, пов’язаних з плануванням маршрутів руху БПЛА. Розглянуто архітектуру з використанням ШІ в СМ ПНО.

Подальші дослідження доцільно спрямувати на розроблення моделей оптимізації кількості літаючих вузлів (БПЛА) у складі підсистем ЛХО, ЛГО та ЛТО за критерієм швидкості обробки моніторингової інформації.

## Література

1. Zaidi S., Atiquzzaman M., Calafate T., Internet of flying things (IoFT): A survey (2020), Computer Communications 2020, 165, 53–74. DOI: 10.1016/j.comcom.2020.10.023.

2. Mahmoud S., Mohamed N., Broker architecture for collaborative UAVs cloud computing (2015), Proceedings of the 2015 International Conference on Collaboration Technologies and Systems (CTS), 2015, pp. 212–219. DOI: 10.1109/CTS.2015.7210423.

3. Mahmoud S., Mohamed N., Al-Jaroodi J., Integrating UAVs into the Cloud Using the Concept of the Web of Things (2015), Journal of Robotics 2015, 2015, 631420. DOI: 10.1155/2015/631420.

4. Sara M., Jawhar I., Nader M., A softwarization architecture for UAVs and WSNs as Part of the cloud environment (2016), Proceedings of the 2016 International Conference on Cloud Engineering Workshops (IC2EW), 2016, pp. 13–18. DOI: 10.1109/IC2EW.2016.17.

5. Majumder S., Prasad M. S., Cloud based control for unmanned aerial vehicles (2016), Proceedings of the 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), 2016, pp. 421–424. DOI: 10.1109/SPIN.2016.7566731.

6. Yapp J., Seker R., Babiceanu R., UAV as a service: Enabling on-demand access and on-the-fly re-tasking of multi-tenant UAVs using cloud services 2016, Proceedings of the 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), 2016, pp. 1–8. DOI: 10.1109/DASC.2016.7778007.

*7.* Youssef S. B. H., Rekhis S., Boudriga N., Bagula A., A cloud of UAVs for the delivery of a sink as a service to terrestrial WSNs (2016), Proceedings of the 2016 14th International Conference on Advances in Mobile Computing and Multi Media (MoMM), 2016, pp. 317–326. DOI: 10.1145/3007120.3007138.

8. Zhang Y., Yuan Z., Cloud-based UAV data delivery over 4G network (2017), Proceedings of the 2017 10th International Conference on Mobile Computing and Ubiquitous Network (ICMU), 2017, pp. 1–2. DOI: 10.23919/ICMU.2017.8330084.

9. Hong C., Shi D., A cloud-based control system architecture for multi-UAV (2018), Proceedings of the 2018 3rd International Conference on Robotics, Control and Automation (ICRCA), 2018, pp. 25–30. DOI: 10.1145/3265639.3265652.

10. Stan R. G., Negru C., Pop F. CloudWave: Content gathering network with flying clouds (2019), Future Generation Computer Systems 2019, 98, 474–486. DOI: 10.1016/j.future.2019.03.033.

11. Rodrigues M., Branco K. R. L. J., Cloud-SPHERE: Towards Secure UAV Service Provision (2020), Journal of Intelligent & Robotic Systems 2020, 97, 249–268. DOI: 10.1007/s10846-019-01046-6.

12. Narang M., Xiang S., Liu W., Gutierrez J., Chiaraviglio L., Sathiaseelan A., Merwaday A., UAV-assisted edge infrastructure for challenged networks (2017), Proceedings of the 2017 IEEE Conference on Computer Communications Workshops (WKSHPS), 2017, pp. 60–65. DOI: 10.1109/INFCOMW.2017.8116353.

13. Cheng N., Xu W., Shi W., Zhou Y., Lu N., Zhou H., Shen X., Air-Ground Integrated Mobile Edge Networks: Architecture, Challenges, and

Opportunities (2018), IEEE Communications Magazine 2018, 56, 26–32. DOI: 10.1109/MCOM.2018.1701092.

14. Zhou Z., Feng J., Tan L., He Y., Gong, J., An Air-Ground Integration Approach for Mobile Edge Computing in IoT (2018), IEEE Communications Magazine 2018, 56, 40–47. DOI: 10.1109/MCOM.2018.1701111.

15. Chen W., Liu B., Huang H., Guo S., Zheng Z., When UAV Swarm Meets Edge-Cloud Computing: The QoS Perspective (2019), IEEE Network 2019, 33, 36–43. DOI: 10.1109/MNET.2019.1800222.

16. Zhou F., Wu Y., Sun H., Chu Z., UAV-Enabled mobile edge computing: Offloading optimization and trajectory design (2018), Proceedings of the 2018 IEEE International Conference on Communications (ICC), 2018, pp. 1–6. DOI: 10.1109/ICC.2018.8422277.

17. Zhou F., Wu Y., Hu R. Q., Qian Y., Computation rate maximization in UAV-Enabled wireless-powered mobile-edge computing systems (2018), IEEE Journal on Selected Areas in Communications 2018, 36, 1927–1941. DOI: 10.1109/JSAC.2018.2864426.

18. Hu X., Wong K.-K., Yang K., Zheng Z., UAV-Assisted Relaying and Edge Computing: Scheduling and Trajectory Optimization (2019), IEEE Transactions on Wireless Communications 2019, 18, 4738–4752. DOI: 10.1109/TWC.2019.2928539.

19. Li J., Liu Q., Wu P., Shu F., Jin S., Task Offloading for UAV-based Mobile Edge Computing via Deep Reinforcement Learning (2018), Proceedings of the 2018 IEEE/CIC International Conference on Communications in China (ICCC), 2018, pp. 798–802. DOI: 10.1109/ICCChina.2018.8641189.

20. Messous M. A., Senouci S. M., Sedjelmaci H., Cherkaoui S., A Game Theory Based Efficient Computation Offloading in an UAV Network (2019), IEEE Transactions on Vehicular Technology 2019, 68, 4964–4974. DOI: 10.1109/TVT.2019.2902318.

21. Nguyen V. D., Khanh T. T., Van Nam P., Thu N. T., Seon Hong C., Huh E. N., Towards Flying Mobile Edge Computing (2020), Proceedings of the 2020 International Conference on Information Networking (ICOIN), 2020, pp. 723–725. DOI: 10.1109/ICOIN48656.2020.9016537.

22. You W., Dong C., Cheng X., Zhu X., Wu Q., Chen G., Joint Optimization of Area Coverage and Mobile-Edge Computing with Clustering for FANETs (2021), IEEE Internet of Things Journal 2021, 8, 695–707. DOI: 10.1109/JIOT.2020.3006891.

23. Lee G., Saad W., Bennis M., Online Optimization for UAV-Assisted Distributed Fog Computing in Smart Factories of Industry 4.0 (2018), Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), 2018, pp. 1–3 DOI: 10.1109/GLOCOM.2018.8647441.

24. Mohamed N., Al-Jaroodi J., Jawhar I., Noura H., Mahmoud S., UAVFog: A UAV-based fog computing for Internet of Things (2017), Proceedings of the 2017 IEEE SmartWorld Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data

Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2017, pp. 1–8. DOI: 10.1109/UIC-ATC.2017.8397657.

25. Ti N. T., Bao Le L., Joint resource allocation, computation offloading, and path planning for UAV based hierarchical fog-cloud mobile systems (2018), Proceedings of the 2018 IEEE 7th International Conference on Communications and Electronics (ICCE), 2018, pp. 373–378. DOI: 10.1109/CCE.2018.8465572.

26. Hou X., Ren Z., Cheng W., Chen C., Zhang H., Fog Based Computation Offloading for Swarm of Drones (2019), Proceedings of the 2019 IEEE International Conference on Communications (ICC), 2019. DOI: 10.1109/ICC.2019.8761932.

27. Devraj, Rao R. S., Das S., Fog Computing Environment in Flying Ad-hoc Network (2022), Cloud Computing Enabled Big-Data Analytics in Wireless Ad-hoc Networks / ed. by S. Das, R. S. Rao, I. Das, V. Jain, N. Singh, Boca Raton, CRC Press, 2022, pp. 31–48. DOI: 10.1201/9781003206453-3.

28. Uddin M. A., Ayaz M., Mansour A., Aggoune, el H. M., Sharif Z., Razzak I., Cloud-connected flying edge computing for smart agriculture (2021), Peer-to-Peer Networking and Applications 2021, 14, 3405–3415. DOI: 10.1007/s12083-021-01191-6.

29. Yazid Y., Ez-Zazi I., Guerrero-González A., El Oualkadi A., Arioua M., UAV-enabled mobile edge-computing for IoT based on AI: A comprehensive review (2021), Drones 2021, 5, 631420. DOI: 10.3390/drones5040148.

30. Koubaa A., Ammar A., Abdelkader M., Alhabash Y., Ghouti L., AERO: AI-Enabled Remote Sensing Observation with Onboard Edge Computing in UAVs (2023), Remote Sensing 2023, 15, 1873. DOI: 10.3390/rs15071873

31. Wang C. Y., Bochkovskiy A., Liao H. Y. M., YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors (2022), arXiv 2022, arXiv:2207.02696. DOI: 10.48550/arXiv.2207.02696.

32. Wojke N., Bewley A., Paulus D., Simple online and realtime tracking with a deep association metric (2017), Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), 2017, pp. 3645–3649. DOI: 10.1109/ICIP.2017.8296962.

33. Shafi O., Rai C., Sen R., Ananthanarayanan G., Demystifying TensorRT: Characterizing Neural Network Inference Engine on Nvidia Edge Devices (2021), Proceedings of the 2021 IEEE International Symposium on Workload Characterization (IISWC), 2021, pp. 226–237. DOI: 10.1109/IISWC53511.2021.00030.

# 11. ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ІНТЕРНЕТУ РЕЧЕЙ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ЦИФРОВИХ ДВІЙНИКІВ

**В. В. Гаєвський[1], Т. В. Кунуп[2], О. І. Морозова[3], В. Р. Щеглов[3]**

*[1]«НВП «Залізничавтоматика»*
*[2]ВСП «Одеський технічний фаховий коледж ОНТУ»*
*[3]Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»*

## 11.1. Вступ

Сучасні виробництва стикаються із необхідністю адаптувати сучасні інформаційні технології до нових викликів і вимог ринку. Революція у виробництві, яка спрямована на досягнення максимальної ефективності через використання автоматизації, аналітики великих даних, прогнозованого обслуговування та IoT, зазвичай називають «промисловою революцією 4.0» (Industry 4.0) або «розумним виробництвом» [1]. Розумне виробництво може виконувати складні задачі без втручання людини, знижуючи витрати та підвищуючи безпеку працівників і обладнання, зменшуючи кількість відходів у середовище, а також уникаючи втручання людини до небезпечних, брудних і важких робіт.

Цифрові двійники (ЦД) разом з інтернетом речей (IoT), аналізом великих даних, машинним навчанням (ML), кіберфізичними системами (CPS), розвиток 4G і 5G можна вважати ключовими факторами, що сприяють промисловій революції 4.0 [2]. ЦД спрямований на створення точної цифрової моделі фізичного об'єкта або процесу, яка здатна збирати інформацію з реального середовища, виконувати перевірку, оцінювання, оптимізацію та прогнозування на цій моделі перед безпечним застосуванням результатів на фізичний об'єкт або процес. Це допомагає у прийнятті рішень в реальному часі для підвищення ефективності роботи, а також пом'якшення або запобігання неочікуваних подій протягом життєвого циклу реального об'єкта [3].

Одна з перших концепцій ЦД була реалізована NASA в рамках програми «Аполлон». На її основі було побудовано два ідентичних космічних апарата – один із них встановився на космічному кораблі, а інший використовувався для віддзеркалення поведінки першого у різних умовах під час місії, щоб заздалегідь прийняти більш точні рішення для управлінням космічним апаратом [4]. Ідею ЦД вперше описав у 1991 році Девід Гелернтер у своїй книзі «Дзеркальні світи». А пізніше, у 2002 році, доктор Майкл Грівз представив і описав концепцію цифрових двійників [5]. Назва «Digital Twin» (DT) була представлена NASA в чернетці технологічної дорожньої карти в 2010 році як система моделювання «для відображення життя його літаючого двійника» [6].

Відтоді технологія ЦД привернула увагу дослідників і стала однією з ключових технологій у сучасному виробництві. У 2020 році розмір світового ринку ЦД оцінювався в 3,1 млрд доларів США [7], а до 2027 року очікується, що він досягне 63,5 млрд доларів США. За прогнозами IDC, з 2021 по 2027 рік кількість нових фізичних активів і процесів, які моделюються як ЦД збільшиться з 5% до 60%, що призведе до оптимізації операційної продуктивності [8]. Зі значним збільшенням числа інтелектуальних пристроїв, підключених до кіберпростору, зростанням популярності та розвитком хмарних обчислень, аналітики великих даних і машинного навчання, інтеграція ЦД стає більш природною та доступною для сучасних виробництв і бізнесу, які можуть зменшити час на розробку продуктів – від концепції до поставки, можуть оптимізувати діяльність, покращити безпеку, мати більше контролю та знизити витрати на обслуговування [6].

Метою цієї роботи є огляд концепції ЦД, аналіз ключових доменів разом із прикладами їх використання, особливостями, проблемами, обмеженнями та перевагами, а також формалізації загальних проблем та обмежень у ЦД.

## 11.2. Огляд видів ЦД

ЦД ще активно розвивається і формуються. Досі не вистачає стандартизації, інструментів, а іноді й успішно реалізованих прикладів у різних галузях, а також чітко визначеної термінології [9]. Оскільки ЦД працюють з гетерогенними системами, а також включають інші дисципліни та технології, їх можна класифікувати за різними аспектами. Розглянемо основні типи класифікацій ЦД.

### 11.2.1. За підходами до створення моделі

Одним із основних інструментів у ЦД є моделювання. Модель дозволяє виконувати обчислення, прогнозувати збої, оптимізовувати поведінку перед застосуванням до фізичного об’єкта, відтворювати певні сценарії у віртуальному просторі.

Існує три основні підходи до моделювання: фундаментальне моделювання, моделювання на основі даних і гібридне моделювання [9].

За допомогою **фундаментального моделювання** можливо створити складну модель відповідно до законів фізики, математики або хімії, вона вимагає менше даних, є більш загальною та може бути застосована до іншого двійника з такими ж фундаментальними характеристиками. Але це вимагає більше обчислювальних ресурсів і може бути неточним через деякі індивідуальні особливості конкретних об’єктів. У деяких випадках не можливо створити або знайти фундаментальну модель, або обчислювальні ресурси для фундаментальної моделі надто великі, або важко створити точну модель для

конкретного об'єкту зі своїм набором індивідуальних характеристик. До того ж часто важко охопити усі фундаментальні закони.

**Моделювання на основі даних (data-driven)** не потребує опису процесів та об’єктів з точки зору законів із прикладних наук. Така модель вчиться на досвіді, який отримує з великої кількості даних - тобто це більш емпіричний досвід. Це позбавляє необхідності описувати модель точними алгоритмами, проте є ризик мати неякісні дані, неякісну архітектуру для машинного навчання. До того ж не завжди є можливість мати постійний потік актуальних великих даних.

Варіантом рішенням згаданних проблем може бути **гібридне моделювання** – це компроміс між двома попередніми варіантами. Наприклад, фундаментальна фізична модель може бути підкриплена якимось досвідом на основі даних.

### 11.2.2. За ієрархію

Переважна кількість оглядових робіт про цифрові двійники містить класифікацію за ієрархією. Іноді у класифікаціях невистачає пунків, або змінені назви. Опишемо більш повну версію такої ієрархії (рис. 11.1) [6].

**Двійники компонентів** (part/component twin) - моделюють найменший повноцінний компонент, що може бути частиною обладнання, процесу, матеріалу. Так клапан або простий датчик може бути таким компонентом

**Двійники активів** (asset twin) - моделюють декілька простих компонентів та зв’язки між ними. А також як кілька компонентів працюють разом як єдине ціле. Прикладом може бути двигун, що складається з кількох компонентів.

**Двійники систем** (system/unit twin) - такі двійники вже поєднують декілька двійників активів, що можуть вже моделювати цілу систему, як наприклад виробнича лінія, цех, літак тощо.

**Двійники процесів** (process twin) - іноді ще називають системою систем, поєднує одразу кілька або усі системи. Надає найбільш широкий погляд на співпрацю систем. Це вже рівень моделювання усієї фабрики.

Таким чином, ЦД можуть представляти як прості датчики та насоси, так і об’єднувати, моделювати кілька виробничих підсистем. Крім того, для моделювання двійників більш високого рівня, не обов’язково моделювати усі компоненти більш низьких рівнів - це впливатиме на точність, але все залежить від задач та ресурсів.

### 11.2.3. За життєвим циклом продукту

ЦД може створюватися відповідно до життєвого циклу продукту [6]. Поділяють цифрові двійники в такій класифікації на:

1. **ЦД прототип (DTP - Digital Twin Prototype)** - моделює продукт, що тільки може бути створений. Це допомогає із попередніми оцінками та аналізом доцільності фізичної реалізації.

2. **ЦД сутність (DTI - Digital Twin Instance)** - це двійник вже існуючого об'єкту. Працює разом із фізичним двійником до завершення работи продукту. Допомогає у моніторингу, передбаченнях та моделюванні можливої поведінки.

3. **ЦД агрегатор (DTA - Digital Twin Aggregate)** - являється представленням усіх продуктів одного типу, що були зроблені. Використовується для загальних тестів та оцінок.

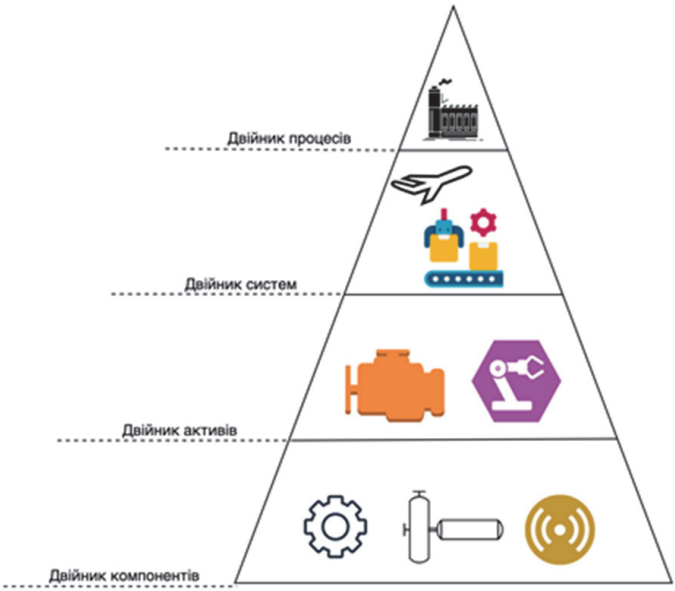



Рисунок 11.1 – Класифікація ЦД за ієрархією

11.2.4. За рівнем інтеграції

Інший погляд на цифрові двійники - це класифікація за рівнем інтеграції. Виділяють наступні типи (рис. 11.2) [6, 10]:

**Цифрова Модель (Digital Model)** - тип ЦД, коли дані між фізичним об'єктом та цифровою моделлю передаються у ручному режимі без автоматичної реакції на події.

**Цифрова Тінь (Digital Shadow)** - інший тип ЦД, де дані від фізичного об'єкту передаються автоматично до віртуального представлення. Але для зворотнього зв'язку все ще необхідно використовувати ручні інструменти.

**Цифровий Двійник (Digital Twin)** - в цьому типі вже наявний двусторонній зв'язок між фізичним та цифровим представленням. Отже, зміни на продукті швидку йдуть в обробку та аналіз. Після чого результати з моделі можуть також автоматично бути застосовані до продукту.

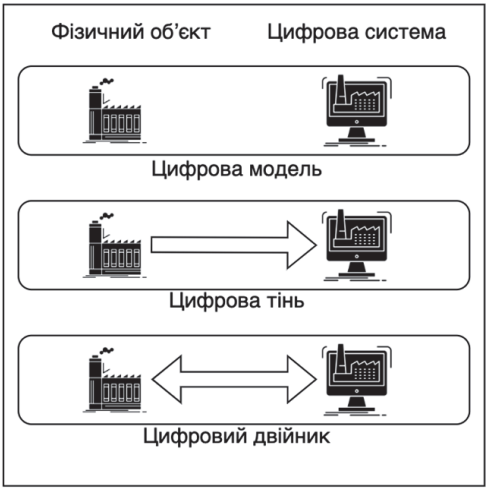



Рисунок 11.2 – Рівні інтеграції

11.2.5. За рівнем зрілості

Цифрові двійники мають певну градацію по розумності або зрілості. Така зрілість залежить від слкадності моделі, якості даних та поставленної задачі, разом з цим зростає і цінність. Розглянемо рівні зрілості ЦД (рис. 11.3) [6]:

**Описовий -** візуальна та інша описова інформація стану продукту, процесу. Це можуть бути різні приборні панелі, графіки, 3д візуалізації.

**Інформативний -** такий ЦД має покращену інтеграцію з даними. Модель допомогає робити певні висновки та знаходити певні шаблони.

**Прогнозний -** історичні та поточні дані надають можливість такому ЦД робити передбачення та прогнозувати певні події, стан, відмови.

**Предписуючий -** покращена версія прогнозного ЦД. На базі зроблених передбачень висуваються певні рекомендації щодо процесів, фізичних представлень двійників.

**Розумний/автономний -** найбільш розвинений тип ЦД, що може працювати в автономному режимі, реагуючи на дані у реальному часі, роблячі передбачення та надаючи відповідні команди до фізіичного двійника.

11.2.6. За необхідним рівнем цифровізації

Для створення повноцінного цифрового двійника, що буде виступати у ролі ЦД сутності (DTI), необхідно мати достатній рівень цифровізації фізичної системи. Тому можна поділити такі фізичні системи або об'єкти на ті, що підходять за рівнем цифровізації, та ті, які ще потребують підготовчих робіт з цифровізації.\

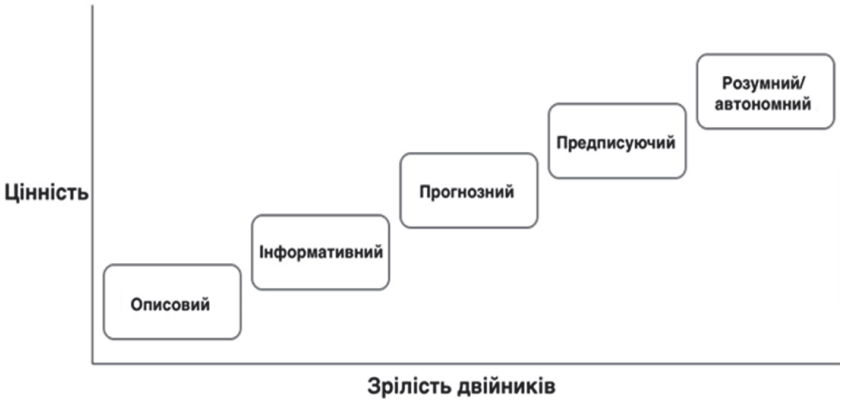



Рисунок 11.3 – Класифікація ЦД за зрілістю

На рис. 11.4 можна побачити умовне розбиття систем на 7 рівнів цифровізації. Перші 3 рівні (відсутність цифровізації, комп'ютерізація, рівень підключення) можна віднести до процесу поступової цифровізації системи, де компоненти можуть використовувати IT технології, але не працювати як розумна система у поєднанні з аналізом людей. Останні 4 рівні вже підходять за рівнем цифровізації - вони характеризуються взаємозв'язком людей, приборів, систем розумним автоматизованим шляхом в єдину систему.

Це певною мірою корелює з класифікацією за рівнем зрілості ЦД, але тут мова саме про цифровізацію - не про віртуальне представлення об'єкту.

11.2.7. За галузями

В цій роботі більше оглядаються та порівнюються ЦД саме у контексті різних сфер та галузей. Їх детальний опис був виділений в окремий параграф 11.4.

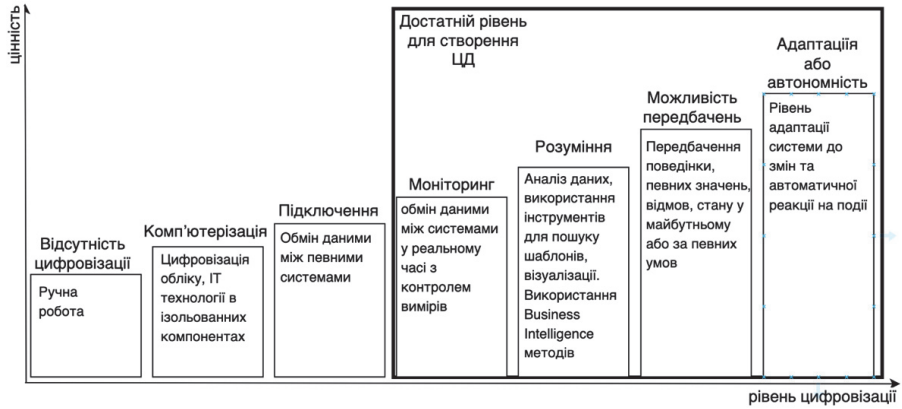



Рисунок 11.4 – Класифікація ЦД за необхідним рівнем цифровізації

11.2.8. Висновки щодо огляду видів ЦД

Було розглянуто декілька основних напрямків класифікацій цифрових двійників. Через гетерогенність, нечітку визначенність деяких термінів та мультидисциплінарність ЦД, кількість класифікацій можуть бути розширена.

Таким чином, з основних напрямків можно умовно побудувати багатовимірний куб класифікацій (приклад рис. 11.5), де кожна умовна комірка буде представляти конкретний набір характеристик - як наприклад, ЦД з предписуючими характеристиками, що є прототипом майбутнього продукту у галузі літакобудування.

11.3. Класифікація джерел

Концепція ЦД може бути застосована майже в усіх сферах діяльності, але в цій статті розглядаються найпоширеніші галузі, які можна вважати основними або які впливали на розвиток цифрових двійників. Таким чином, було переглянуто низку оглядових робіт щодо ЦД, щоб зрозуміти найбільш обговорювані домени з точки зору (таблиця 1.1).

Виробництво, розумні міста та галузі охорони здоров'я можна знайти майже в кожній оглядовій статті чи роботі ЦД. Таблиця 11.1 це підтверджує. Автомобільна та аерокосмічна промисловість завершують п'ятірку найбільших за популярністю ЦД галузей. Це галузі були обрані основними для огляду для поточної статті. Ще три галузі будуть розглянуті через цікаві роботи та для поширенню контексту при аналізі індустрії ЦД, а саме освіта, будівництво та залізничний транспорт.

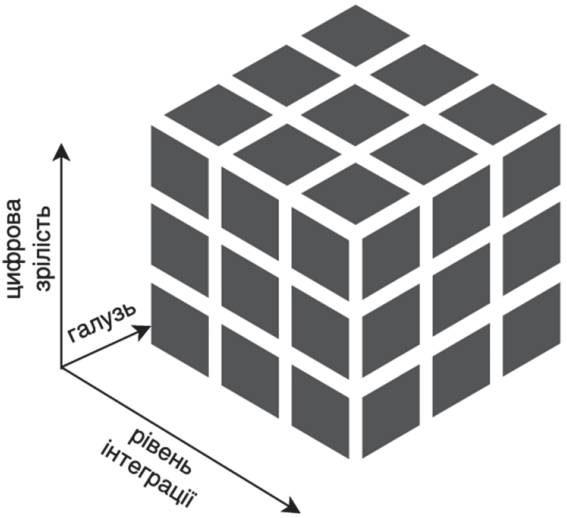



Рисунок 11.5 – Класифікація ЦД за зрілістю

Було розглянуто додаткові роботи із конкретними прикладами впровадження ЦД у кожній із індустрій. Повний перелік галузей, які будуть розглянуті, і пов'язані роботи можна знайти в таблиці 11.2.

Таблиця 11.1 – Галузі та оглядові роботи, в яких ці галузі аналізувалися

| № | Галузі | Оглядові роботи, в яких ці галузі аналізувалися |
|---|---|---|
| 1 | Аерокосмічна | [12, 3, 6] |
| 2 | Автомобільна | [12, 6, 13] |
| 3 | Виробництво | [12, 3, 6, 10, 2, 13, 11] |
| 4 | Гірничодобувна | [12] |
| 5 | Морський транспорт | [12, 3] |
| 6 | Енергетика | [12, 10] |
| 7 | Паливна | [12] |
| 8 | Сільське господарство | [12, 6] |
| 9 | Розумні міста | [12, 3, 6, 10, 13, 11] |
| 10 | Освіта | [12, 6] |
| 11 | Медицина | [12, 3, 6, 13, 11] |
| 12 | Торгівля | [12] |
| 13 | Будівництво | [12] |
| 14 | Транспорт | [10, 13] |

## 11.4. Аналіз джерел та підходів за різними індустріями ЦД

Для визначення переваг і проблем у ЦД, а також формульовані загальних викликів і особливостей, було прийнято рішення проаналізувати ключові й деякі відносно нові індустрії по використанню ЦД.

Це дозволить більш широко дивитися на можливості технології та перевикористання підходів із різних доменів ЦД. У цьому розділі будуть детально розглянуті та проаналізовані обрані галузі ЦД. Кожен із параграфів по галузі містить загальний опис, особливості цього домену, проблеми та приклади використання.

Таблиця 11.2 – Перелік індустрій для огляду та аналізу

| # | Галузь | Роботи за напрямками |
|---|---|---|
| 1 | Виробництво | 1. Оглядові статті, що включають галузь [2, 3, 6, 10-13]<br>2. Discovering the Digital Twin Web – From singular applications to a scalable network [14]<br>3. Towards a Cyber-Physical Manufacturing Cloud through Operable – Digital Twins and Virtual Production Lines [15]<br>4. A Digital Twin for the Logistics System of a Manufacturing Enterprise Using Industrial IoT [16]<br>5. Digital representations of physical assets [17]<br>6. An Integrated Mobile Augmented Reality Digital Twin Monitoring System [18]<br>7. The Fundamental Approach of the Digital Twin Application in Railway Turnouts with Innovative Monitoring of Weather Conditions [19] |
| 2 | Автомобільна | 1. Оглядові статті, що включають галузь [6, 12, 13]<br>2. Traffic Safety Detection System by Digital Twins and Virtual Reality Technology [20]<br>3. Automotive overview – Top 5 Use Cases of Digital Twin in Automotive Industry in 2022 [21] |
| 3 | Медицина | 1. Оглядові статті, що включають галузь [3, 6, 11-13]<br>2. Best Digital Twin Applications & Use Cases in Healthcare in 2022 [22]<br>3. A Novel Cloud-Based Framework for the Elderly Healthcare Services Using Digital Twin [23] |

Продовження табл. 11.2

| | | |
|---|---|---|
| 4 | Аерокосмічна | 1. Оглядові статті, що включають галузь [3, 6, 12]<br>2. UAV, Using Digital Twins and Drones to Capture Physical Environments [7]<br>3. Structural Digital Twin of the UAV that was used to monitor vehicle structural health and drive dynamic flight planning decisions [24]<br>4. Shaun Waterman. Air Force Goes All in on Digital Twinning – for Bombs As Well As Planes [25] |
| 5 | Розумні міста | 1. Оглядові статті, що включають галузь [3, 6, 10-13]<br>2. The Digital Twin of the City of Zurich for Urban Planning [26]<br>3. A systematic review of a digital twin city: A new pattern of urban governance toward smart cities [27]<br>4. Smart Cities with Digital Twin Systems for Disaster Management [28]<br>5. City Digital Twin Potentials: A Review and Research Agenda [29] |
| 6 | Освіта | 1. Оглядові статті, що включають галузь [6, 12]<br>2. Development of a digital twin of a flexible manufacturing system for assisted learning [30]<br>3. Tangibles and Digital Twins: Toward Meaningful Learning Support in Cyber- Physical System Development [31] |
| 7 | Будівництво | 1. Оглядові статті, що включають галузь [12]<br>2. Building Lifecycle Management, Cognitive Digital Twins [32]<br>3. Options for and Challenges of Employing Digital Twins in Construction Management [33] |
| 8 | Залізничний транспорт | 1. Оглядові статті, що включають галузь [10, 13]<br>2. Digital twins for managing railway maintenance and resilience [40]<br>3. Digital Twins in Railways [41]<br>4. The Fundamental Approach of the Digital Twin Application in Railway Turnouts with Innovative Monitoring of Weather Conditions [42]<br>5. Towards a Data-driven Operational Digital Twin for Railway Wheels [43] |

### 11.4.1. Виробництво

Сьогодні виробництво є найпопулярнішою сферою для прикладів із впровадження та інтеграції цифрових двійників. Цей сектор включає різні формати використання ЦД на кожному із етапів життєвого циклу продукту – від проектування та створення прототипу до виробництва, логістики та обслуговування. ЦД може допомогти з прогнозуванням відмови обладнання, контролем точності та продуктивності виробництва, оцінювати концепції перед їх впровадженням, а також покращувати досвід користувачів через візуалізацію [3]. Очікується, що ЦД стане головним інструментом в MBSE (model-based systems engineering) або системній інженерії на основі моделі, оскільки ЦД можна застосовувати на кожному етапі життєвого циклу системи/продукту. Цілком можливо, що вироблені продукти матимуть свої ЦД, які допоможуть у забезпеченні ефективної підтримки продукту та персоналізованого обслуговування та досвіду використання [6].

Одним із викликів у виробничих системах є автоматизація та масове виробництво економічно ефективним способом. ЦД може допомогти в оцінці та аналізі ефективності виробництва, а також кожного із факторів проектування на етапах життєвого циклу.

В роботі [2] було запропоновано модель із 3 типами застосувань у цехах, які фокусуються на різних аспектах виробництва:

- ЦД продукту – вихід або результат виробничого процесу;
- ЦД процесу – робочий процес виробничої лінії, як працює процес;
- операційні ЦД – моніторинг, операційні процедури та контроль.

В іншій роботі пропонується додати ще 4-й пункт до розглянутої класифікації – утилізація та переробка відходів [12].

У дисертаційній роботі [15] пропонується працездатна модель ЦД кіберфізичної виробничої хмари, яка успадковує функції моніторингу із хмари, що відкриває двері для майбутніх виробничих систем. В рамках дисертаційної роботи [14] були побудовані ЦД, орієнтовані на дизайн виробничого обладнання, зокрема на промисловому мостовому крані. У роботі [17] було представлено набір інструментів FA3ST (Fraunhofer-advanced AAS tools for digital twins) як репрезентативну реалізацію загальної та гнучкої архітектури для управління ЦД, яка фокусується на функціональності на завершальній стадії виробництв.

ЦД також можуть бути корисними під час навчання інженерів або для допомоги та контролю низькокваліфікованих операторів. Операторам потрібні глибокі знання обладнання, щоб швидко приймати важливі рішення під час технічного обслуговування або виробничих процесів. У статті [18] пропонується використовувати мобільну систему віддаленого моніторингу доповненої реальності, щоб допомогти операторам із низькою кваліфікацією полегшити робочий процес. Таку систему було побудовано для баштового крана.

Логістика – ще одна проблема сучасного виробництва. У статті [16] автори розглядають застосування ЦД для логістичної системи виробничого підприємства з використанням IIoT. Також було розглянуто кейс для ПрАТ «ФЕД». Система була розроблена з метою розрахунку оптимального розташування виробничих потужностей для максимізації продуктивності на виробництві.

ЦД у виробництві мають власний перелік проблем:

– проблема роботи ЦД у режимі реального часу – виробництво вимагає швидкої реакції. В той же час ЦД потребує потужності для моделювань та передбачень

– великі дані – величезна кількість субдоменів та взаємозв'язків ускладнює систему та видає великий потік даних.

### 11.4.2. Автомобільна галузь

Сучасні автомобілі вже оснащені великою кількістю телеметричних датчиків, що генерують величезну кількість даних для подальшого аналізу – таким чином сучасні авто вже більше менш відповідають необхідному рівню цифровізації та готові для впровадження ЦД. Проте це лише частина системи ЦД, і цей сектор активно розвивається. Переважна кількість існуючих робіт зосереджено на процесах виробництва та прототипування, а також валідації характеристик майбутніх автомобілів [13]. Більшість рішень із виробничого сектору ЦД можна також застосувати і в автомобільному виробництві, але автомобільна промисловість має свої особливості. Виробники можуть скористатися функціями прогнозованого технічного обслуговування. Це стосується не тільки виробничих роботів, а й технічного обслуговування автомобілів [6, 12, 21]. ЦД можуть зменшити витрати та кількість помилок шляхом застосування віртуального тестування для перевірки автомобіля [13]. Механіки можуть надавати більш швидкі та персоналізовані рішення за коротший час.

Службам прокату буде легше слідкувати за станом автомобіля у реальному часі [6]. ЦД також використовується для отримання більш персоналізованого досвіду водіння автомобіля – виробники можуть аналізувати поведінку водія та налаштовувати під них автомобілі або надавати індивідуальні функції, які задовольняють потреби клієнтів [12]. Іншим фактором є покращення досвіду продажів – покупці можуть перевірити продукт у різних конфігураціях та поведінку за допомогою симуляції, інструментів VR та AR, що стимулюють продажі через створення позитивного досвіду користувача [12, 21].

Деякі сучасні автомобільні компанії вже інтегрували ЦД у свій бізнес. Tesla Motors використовує ЦД у кожному автомобілі. Це дозволяє швидко оновлювати автомобіль відповідно до виявлених індивідуальних проблем, як наприклад, віддалено компенсувати проблеми з дверима шляхом налаштування

гідравліки [12]. Volkswagen використовує ЦД з одним зі своїх роботів при плануванні виробництва. Це економить близько трьох тижнів часу та 40 квадратних метрів виробничої площі [12]. Maserati також інтегрувала ЦД у свій виробничий процес, щоб оцінити, як зміни автомобіля впливають на виробничі процеси [12]. Крім того, у роботі [20] досліджується прогнозування типів водіння транспортного засобу для підвищення точності визначення безпеки руху.

Величезні дані, які генеруються автомобілями, можна вважати одним із головних викликів. Підраховано, що компанії аналізують лише 12% доступних даних [21]. Це означає, що ефективну модель неможливо побудувати з такою низькою швидкістю обробки даних. Крім того, сам автомобіль може бути оффлайн, тому необхідно обробляти це відповідним образом. Іншою проблемою є безпека – атака на ЦД автомобілів на сервері, що впливає на рішення, або навіть керує автомобілем, може поставити під загрозу процес водіння в цілому.

### 11.4.3. Медицина

У секторі охорони здоров’я ЦД можуть покращити ефективність організації, забезпечити більш персоналізоване лікування та контроль. Рішення на основі ЦД у медицині можуть підвищити точність аналізу стану організму та встановлення діагнозу, допомогти з розподілом медичних ресурсів та хірургічним асистуванням, лікарі можуть призначати плани лікування на основі даних у реальному часі. Для пацієнтів ЦД може контролювати та аналізувати стан їхнього тіла та надавати зворотний зв’язок у режимі реального часу.

ЦД в медицині можна розділити на три основні категорії [22]:

1) **ЦД закладу охорони здоров'я.** ЦД можна використовувати для створення лікарні-двійника, щоб зрозуміти, як справлятися з труднощами в різних ситуаціях, наприклад, пожежа в будівлі або надмірна потреба в травматичних кабінетах. Крім того, ЦД може допомогти з управлінням активами та обладнанням, як в випадку, щоб зрозуміти, чи є дефіцит ліжок чи іншого обладнання під час пандемії в поточний момент;

2) **ЦД організму людини.** Віртуальні двійники тіла або його компонентів забезпечують більш детальне уявлення про фізіологічний стан, що дозволяє точніше та персоналізовано діагностувати хвороби та планувати лікування. Крім того, такі рішення можуть допомогти виявити патології та інші захворювання ще до того, як порушення стануть очевидними – це приклад прогнозного лікування;

3) **ЦД для медицини та розроблення пристроїв.** На етапі розроблення нового медичного пристрою завжди важливо оцінити та перевірити, як він працює та чи відповідає певним критеріям. ЦД можна використовувати для практичного тестування нових пристроїв або функцій перед їх використанням. Ту саму концепцію можна застосувати й до ліків –

віртуальні біохімічні моделі ліків можуть допомогти вченим у модифікації або розробленні нових методів лікування.

Концептуальну модель людського ЦД або human digital twin (HDT) було розглянуто в [13]. Ідея полягає в тому, щоб відтворити тіло людини в кіберфізичному просторі. Дані з датчиків і медичних карт постійно аналізуються та використовуються для надання більш персоналізованого лікування або моніторингу стану організму. Робота [13] розглядає деякі приклади HDT, такі як вимірювач серцебиття, вимірювач кроків та ін. SmartFit допомагає з персоналізованими порадами та відстеженням стану тіла. Іншим застосуванням є хірургічне планування та виконання [11]. ЦД були використані для створення «Cardio Twin» для запобігання ішемічній хворобі серця та інсульту [6]. Стаття [23] пропонує нову концепцію ЦД охорони здоров'я, яка була створена для впровадження таких послуг, як моніторинг у режимі реального часу для людей похилого віку. Крім того, в цій роботі [23] стверджує, що дослідження моделювання охорони здоров'я зосереджені переважно на освіті в галузі охорони здоров'я, а більшість досліджень ЦД у медицині зосереджено на платформах для моніторингу.

Сфера охорони здоров'я має свої проблеми. Однією з найпоширеніших проблем є безпека даних. Під час лікування, аналізів збирається значна кількість індивідуальних даних, які слід зберігати та обробляти безпечним способом. У деяких випадках лише клініки можуть зчитувати дані пацієнтів, а це означає, що моделювання та прогнозування можна виконувати лише локально в клініках або із обфускованими та анонімоманими даними. Крім того, якість даних і злиття даних (data fusion) – це більш загальні проблеми, актуальні для всіх напрямів ЦД [22]. З медичної точки зору іноді біологічні явищ погано вивчені або їх важко змоделювати, наприклад проблема моделювання емоцій. Тому експертна валідація моделей є дуже важливою для оцінювання надійності моделей, а також для якості їх роботи [13]. Ще одна досить цікава тема, яка може бути проблемою, це проблема соціальної етики [13]. Наприклад, як сприймають люди факт масової цифровізації внутрішніх частин і процесів тіла для проведення подальших моделювань?

### 11.4.4. Аерокосмічна галузь

ЦД також використовуються в аерокосмічній промисловості. Аерокосмічні компанії почали використовувати ЦД, щоб мінімізувати час простою двигунів і компонентів, використовують переваги прогнозованого технічного обслуговування, а також використання віртуального моделювання для перевірки поведінки літаків і кораблів в різних середовищах та умовах, наприклад, в екстремальних погодних умовах [3]. ЦД можна використовувати на етапі виробництва для розроблення та тестування продуктів. Моделювання та тестування з точки зору оптимізації продуктивності також є місцем для використання ЦД [12].

Переваги можна формалізувати таким чином:

– безпечніші місії, максимізується точність та їх успішність;

– дешевші космічні апарати, менші витрати на експлуатацію та обслуговування;

– перевірка продукту та його моделювання перед виробництвом.

NASA розробило ЦД ракетного двигуна для прогнозування умов польоту, впливу цих умов і способів оптимізації запуску двигуна [12]. Крім космічних кораблів і літаків, безпілотні літальні апарати (БПЛА) можуть розглядатися як предмет інтеграції ЦД в рамках аерокосмічної сфери. Але сам БПЛА часто використовується як інструмент для впровадження нових ЦД. Наприклад, робота [7] містить огляд того, як БПЛА можна використовувати для захоплення фізичного середовища та зйомки ЦД дахів, зйомки полів, щоб аналізувати стан рослин. [24] провели дослідження із використання ЦД БПЛА для моніторингу його справності та динамічної зміни рішень щодо планування польоту відповідно до стану та рекомендацій моделі машинного навчання – такий концепт використання масштабується і на літаки з космічними кораблями.

Ще один спосіб використання ЦД в аерокосмічній галузі – це моделювання та випробування нових кораблів перед вибором найкращого варіанту для виробництва. Наприклад, реактивний літак повітряних сил США eT-7 Red Hawk був розроблений і випробуваний з використанням концепції ЦД [25]. Тобто валідація концепту із конкурентами та оцінювання його ефективності були ще проведені до моменту виробництва прототипів.

Аерокосмічну галузь можна віднести до систем реального часу або критичної інфраструктури. Тому вимоги до безпеки та відклику мають бути на високому рівні.

Одним із викликів аерокосмічної галузі є комунікації. ЦД вимагають величезної кількості даних для передачі з кораблів у хмару та у зворотному напрямі. Тому обговорюється можливість застосування 5G для зв'язку. Через значний обсяг даних, тут зустрічаються й відповідні загальні проблеми в обробленні даних. Крім того, відсутність відповідних знань про проектування ЦД, інструментів і стандартизації може бути проблемою для інтеграції ЦД в аерокосмічну галузь.

### 11.4.5. Розумні міста

Міста стають розумнішими та вони виробляють інформацію з різних джерел, таких як вуличні камери, міський транспорт, управління водопостачанням, розумні світлофори тощо. ЦД у індустрії містобудування в основному зосереджені на покращенні середовища та якості життя громадян, мобільності та доступності послуг для громадян, безпеці, та щоб технологічний прогрес був орієнтований на людей [3, 13]. Подібно до виробництва, розумні міста включають широкий спектр субдоменів – від транспорту до міського планування. Близько 118 міст використовували ЦД у проектах Smart City [10].

Оскільки ЦД у місті часто використовуються для фізичного моделювання, можна виділити наступні технології, які дозволяють будувати такі моделі: фотографія зі зсувом та нахилом, безпілотний літальний апарат (БПЛА), 3D лазерне сканування та система глобального позиціонування (GPS) [27]. Ще один важливий технологічний фактор – 5G зв'язок, що допомагає з'єднати в одну мережу різні частини розумного міста [27].

**ЦД для управління і соціального контролю міста.** Через цифровізацію процесів, збір даних та можливість моделювати різні сценарії, ЦД стають важливим інструментом в управлінні міста. Влада міста може приймати оптимальні рішення, опираючись на дані, отриманні з моделі. Крім того, відкриті дані та моделі міст є в тому числі й інструментом соціального контролю або демократії, оскільки в такому випадку кожна людина може перевірити забруднення повітря біля будинку, чому в її районі більше ДТП, або чому температура поверхні вулиці (будівлі, дороги) значно більша за сусідню вулицю, що може потім переходити у конкретні запити або дії по усуненню проблеми. Робота [27] розглядає концепцію ЦД міста і те, як вона може змінити структуру та правила управління містом.

**ЦД міста для безпеки та досліджень.** ЦД можуть будувати моделі руху людей і автомобілів, щоб знайти рішення для надзвичайних ситуацій і підвищити ефективність транспортної системи. 3D-моделювання та твіннінг (створення двійників) будівель можуть допомогти створювати якісні моделі якості повітря та моніторингу або прогнозування температури повітря в різних частинах міста у відповідь на погодні умови, стихійні лиха чи різні міські сценарії [3]. Усе це дозволяє підвищити громадську безпеку, покращити мобільність громадян, зменшити витрати та викиди вуглекислого газу, а також допоможе пом'якшити негативний вплив різноманітних аварій та катастроф [10]. Так одна із зон використання «розумного міста» – це стихійні лиха, які негативно впливають на громадські системи, завдаючи шкоди інфраструктурі, переміщуючи населення та порушуючи окремі системи та їхню взаємодію. В роботі [28] розглядаються ЦД розумного міста для моделей управління катастрофами. Також, у роботі [12] зазначено, як ЦД району Доклендс у Дубліні був побудований для прогнозування повеней і попередження людей про можливі повені.

В той же час, ЦД можна використовувати для планування та моделювання типових процесів у місті – від кількості туристів до прогнозування температури. В роботі [26] автори описали ЦД міста Цюріха, що використовується від дослідження урбанізації віртуального туризму та дослідження багатоповерхівок до симуляції шуму на вулицях, забруднення повітря та випромінювання мобільних телефонів. У 2018 році Національний дослідницький фонд Сінгапуру створив віртуальну модель міста, що поєднує 3D-карти та платформу даних із деталями про текстури, будівельні матеріали, геометрію та інші компоненти [12]. Вважається, що таке моделювання буде корисно урядам, громадянам і дослідникам для управління. Міста швидко

змінюються, вони постійно розвиваються, і такі технології можуть не тільки покращити якість життя та безпеку, але й зробити життя більш зрозумілішим, оскільки кожен може отримати доступ до різних симуляцій міста, даних із відкритою статистикою.

З точки зору викликів, домени розумних міст пов'язані з проблемами даних: одна з найбільших проблем – це складність і оброблення даних. Розумне місто являє собою гетерогенну систему з широким спектром підсистем і субдоменів, які потребують високого рівня інтеграції. Інша проблема полягає в точності побудови моделі, відсутності деталей для складного моделювання в рамках усього міста за різними аспектами. Також, через те, що місто є середовищем, в якому живуть люди, виникає проблема приватності через надмірну цифровізацію всіх аспектів міського життя.

### 11.4.6. Освіта

Останнім часом активніше в галузі освіти почали застосовувати ЦД. Основними інструментами є симуляції та технології AR, VR. Вони можуть покращити досвід навчання, візуалізуючи різні аспекти процесів та надаючи можливість взаємодії із змодельованою системою, що призводить до кращого розуміння. Крім того, це може зменшити витрати. Ось деякі з переваг ЦД в галузі освіти:

- покращена доступність – можна отримати віддалений доступ до обладнання;
- вирішує проблему з обмеженими ресурсами, оскільки кожен студент може мати персональне віртуальне обладнання;
- можливість автоматизації системи оцінювання та отримання зворотного зв'язку в режимі реального часу;
- поліпшення досвіду навчання, покращена мотивація та інтерес до процесу навчання;
- безпека для учнів та обладнання;
- здатність навчатися у віртуальних умовах, які важко організувати в реальному житті.

Більшість робіт, які пов'язані із ЦД, згадують освіту як одну з проблем бізнесу чи виробництва. Наприклад, допомога малодосвідченим операторам або підготовка інженерів до майбутньої роботи. У дослідженні [30] пропонується використання ЦД як альтернативної навчальної платформи для курсів виробничої техніки. Автори зосереджені на високоякісному відтворенні фізичної системи, що покращує інформативність від спостереження за процесами та більш детального вивчення учнями. Робота [31] представляє спосіб розробки міждисциплінарної інженерної програми, обговорюючи спільні риси між розробкою CPS (Clicks Per Second) тестів та освітою для розвитку. Подібний підхід був використаний для створення тестового стенда ЦД для

студентів, які можуть досліджувати складності та поведінку системи, взаємодіючі з цим ЦД інженерної системи [12].

Одним із викликів застосування ЦД в освіті є точність відтворення та моделювання процесів та обладнання, щоб студенти могли зрозуміти, як воно працює з необхідним рівнем деталізації, уникаючи помилкового досвіду навчання. Крім того, важко оцифрувати деякий досвід навчання, як наприклад ручне навчання, або роботи в команді. ЦД в освіті – це чудовий інструмент для підвищення продуктивності, мотивації та безпеки студентів під час навчального процесу.

### 11.4.7. Будівництво

Будівництво – це трудомістка та затратна за часом галузь. Крім того, генерується величезна кількість даних у процесі будівництва – від проектування та планування до фізичного розроблення будівлі та її перевірки. Сучасні проєкти стають все більшими та складними, ними стає все важче керувати, до того ж сучасні будівлі часто необхідно перевіряти на енергоефективність, безпеку, інтегрувати автоматичні системи вентиляції, опалення, встановлювати різні датчики та інші пристрої. Все це необхідно вміти спроектувати, побудувати та розмістити значну кількість даних у потрібному місці, обробляючи їх та отримуючи користь від цих даних.

Підходи із ЦД можуть допомогти у більш чіткому контролі та валідації якості процесів будівництва для команд інженерів, а також наддасть можливість заздалегідь та більш комплексно оцінити хід робіт, та деякі особливості зацікавленим сторонам, як наприклад інвесторам або майбутнім клієнтам.

Як і у виробництві, так і у будівельній галузі ЦД можуть використовуватися на різних етапах життєвого циклу проєкту. Використання ЦД може допомогти на етапі прототипування для перевірки та тестування проєктів, це може перевірити деякі особливості, стійкість і поведінку у різних умовах ще перед початком будівництва.

Крім того, це може допомогти безпечно застосувати зміни до проекту в середині будівництва через деякі термінові ситуації [12]. Після будівництва існуючу модель можна використовувати для посилання у разі надзвичайних ситуацій, для підтримки будівлі у ході експлуатації тощо. Часто ЦД є частиною вже існуючих Building Information Modeling (BIM) систем.

**ЦД для безпеки й контролю процесів будівництва.** Безпека й контроль йдуть нерозривно із життєвим циклом будівництва. В багатьох роботах пропонується поєднати у єдину систему управління сам проєкт, а також оцінювання й контролю якості. Так, згідно з роботами, розглянутими в [33], ЦД можуть покращити планування будівельних проектів, здатність швидко вирішувати конкретні потреби та проблеми, такі як фактори вібрації, коливання температури, прогнозувати поведінку у несподіваних подій, допомагати

менеджерам у прийнятті рішень та контролі у ході будівництва та експлуатації або зменшити ризики. В роботі [32] автори досліджують застосовність, сумісність та інтегрованість адаптованої моделі когнітивного ЦД для управління життєвим циклом будівлі. В іншій пов'язаній роботі розглядається література з управління будівельними проєктами через призму ЦД [33]. Автори пропонують триетапну структуру для аналізу та контролю за розвитком ЦД у будівельному секторі. У роботі [39] було запропоновано модель використання ЦД для контролю процесів на об'єкті – сенсори, візуалізація поєднуються із моделлю для автоматичної синхронізації будівельних робіт та підвищення безпеки.

Будівельний сектор стикається з тими ж проблемами, що й виробництво та сектори розумних міст. Багато концепцій взято з цієї області для вирішення проблем будівництва. Але будівництво часто нерозривно йде із досвідом майбутніх користувачів, стейкхолдерів, що потребує якісних інструментів контролю та візуалізації, то ж цей сектор, частіше використовує AR, VR технології, разом із різними симуляціями для покриття цих потреб. Застосування нових технологій, таких як ЦД, може підвищити безпеку, зменшити витрати на будівництво, покращити планування і контроль, відкриваючи нові можливості для більш складних проектів, як ще більші мости, дамби та тонелі.

### 11.4.8. Залізничний транспорт

Зі зростанням населення збільшується й попит до стійких рішень та ефективного управління транспортними системами. Залізниця вважається екологічно чистим («зеленим») видом транспорту порівняно з іншими, як автомобілі, літаки, кораблі тощо. Із активним просуванням концепції збалансованості (sustainability), підвищується й попит до екологічно чистих перевезень. Для контролю вибросів, підвищення ефективності потоків пасажирів та транспорту, прогнозування збоїв та контролю ресурсів і обладнання, необхідно підвищити рівень цифровізації системи у цілому, а також запроваджувати сучасні технології як ЦД.

Як і у багатьох секторах ЦД у залізничній сфері може бути застосований на різних етапах життєвого циклу (проектування, реалізація, експлуатація).

На **етапі проектування** актуальні вже розглянуті схожі підходи – це створення моделі, тестування та валідація деяких аспектів ще до початку виробництва. Наприклад, інструменти Bentley OpenRail від Siemens дозволяють проектувати залізнодорожні проєкти виключно у цифровому форматі, створюючи не тільки 3D та CAD моделі, а й моделі фізичних активів для моніторингу, та перевірки гіпотез. Або DigitalTrains інструмент від DVRS, що дозволяє створювати моделі залізнодорожнього полотна із пересувним составом, моделювати поведінку на поворотах або при використанні упорів для зупинки транспорту.

**Етап реалізації** проєкта тісно пов'язаний із попереднім, та може часто використовувати єдину систему. Так, наприклад, у роботі [40] пропонується використовувати підходи BIM (Building Information Modeling) систем у поєднанні із ЦД для чіткого контролю, верифікації та оптимізації процесів на будівництві об'єкту. Крім того, таке середовище покращує можливості для співпраці між командами та відділами через єдине віртуальне середовище.

Тепер розглянемо основні напрямки у **етапі експлуатації** [41]:

– технічне обслуговування, моніторинг та безпека – прогнозне обслуговування та моніторинг поїздів, залізничного полотна, обладнання, вузлів та ін.;

– планування трафіку – управління потоками транспорту та пересування пасажирів/вантажів;

– управління та політика – управління фізичними активами (рухомий склад та інфраструктура), загальні бізнес процеси на станції;

– обслуговування пасажирів – аналіз поведінки пасажирів для покращення досвіду користувача у період очікування та посадки, додаткової інформативності та навігації по станції.

**ЦД для безпеки залізничного транспорту**. Кожен аспектів важливий, в той же час у роботі [40] після аналізу джерел за напрямком вказано, що більшість наукових робіт виконано саме у аспекті безпеки та підтримки роботи залізнодорожніх шляхів, вагонів та обладнання. Три основні фактори сприяють виникненню дефектів на найбільш вразливих компонентах як залізничні компоненти, стрілочні переводи та провідні рейки залізничних колій і залізничного составу:

– несподівані ситуації, що виникають під час експлуатації та обслуговування;

– вплив навколишнього середовища;

– стирання та ерозія, що спричиняє пошкодження.

Саме тому більшість робіт сфокусована або на створенню ЦД для залізнодорожніх вузлів, щоб виконувати прогнозне передбачення та моніторинг, або схожі дії тільки до критичних компонентів пересувного составу як колеса, трансмісія та зчіпні пристрої. Так у роботі [42] було проаналізовано та створено ЦД для залізничних стрілок, що збирає дані про навколишнє середовище, температуру, вібрації та стан самого компонента. Все це використовується у моделі для прогнозного обслуговування та моніторингу критичного вузла. Іншими прикладом [43] може бути схожа реалізація ЦД для коліс составу, що прогнозує відмови, або попереджає у разі знайдених аномалій у даних.

## 11.5. Практичне застосування ЦД

У подальших роботах практичне застосування ЦД з точки зору інструментів, фреймворків та дослідження технології буде виконувитися у

галузі залізничного транспорту. Залізниця являється важливою частиною інфраструктури, соціальним, економічним та військовим інструментом, а також вважається більш зеленим транспортом. Рішення на базі цифрових двійників у цій індустрії тільки розвиваються, тому актуальним є дослідження та розроблення методик, фреймворків та інструментів.

Сучасні залізничні дороги електрифікуються, впроваджуються більш ефективні локомотиви та інше обладнання, цифровізуються системи керування і моніторингу. В той же час з точки зору безпеки та обслуговування, переважна більшість транспортних подій в Україні на залізниці все ще є наслідком порушення технології виконання робіт та невиконання робіт, що передбачені планами [44]. Серед більш детальних причин слід виділити людський фактор, виснаження систем, що не мають відповідного моніторингу, низька ефективність диагностування та виявлення дефектів, та помилки проектування і будування.

Так, Європейський стандарт EN 50126 CENELEC надає важливий в европейському залізничному транспорті метод оцінки надійності, готовності, ремонтоздатності та безпеки - RAMS. В той же час на показники RAMS найбільш вагомий вплив здійснює технічне обслуговування (ТО) [44], яке є одним з центральних компонентів технологічного комплексу пристроїв та систем залізничної автоматики (ЗА).

Робота [44] була посвячена підвищенню безпеки руху через удосконалення процесу технічної експлуатації систем мікропроцесорної централізації у ЗА. Це робилось шляхом покращення системи оперативної ідентифікації, оцінки й локалізації порушень. Далі, у цьому розділі буде розглянута запропонована модель і архітектура системи мікропроцесорної централізації у ЗА, яка використовує цифрові двійники для покращення показників ідентифікації проблем, їх оцінки, а також передбачення.

**Місце ЦД у системі мікропроцесорної централізації у ЗА.** Перед цифровим двійником в цій системи можна поставити наступну задачу - покращення контролю якості і планування робіт по ТО, а також можливість предективної підтримки - тобто, передбачення потенційних відмов заздалегіть.

У роботі [44] вже вирішувалися подібні проблеми через інший шлях - ідентифікація, оцінка і локалізація порушень та блокування дестабілізаційних факторів. Тобто це покращення інформативності системи, додатковий аналіз для реактивного підходу (тобто операційного) усунення порушень. Ключевим є оперативний персонал, що з допомогою інформованості системи та власного аналізу приймає певні рішення.

За допомогою ЦД пропонується розширення інформативності системи для покращення та прискорення, або навіть превентивних дій персоналу. Для цього інформація про стан системи, буде додатково передаватися до цифрового двійника, який на базі поточних даних, історичних даних та наявної моделі і алгоритмів, буде виконувати незалежний аналіз для надання рекомендацій або передбачень щодо стану системи (рис. 11.6). Таким чином, оперативний

персонал крім поточних даних, буде ще мати додатковий інструмент - систему підтримки прийняття рішень.

**Впровадження пілотного проекту.** Проблема впровадження полягає в тому, що перевірка концепту в готовому вигляді може вимогати великі кошти та займати суттєвий проміжок часу. Крім того, не завжди є можливість зупинити поточну систему для модернізації і перевірки коцепту. Або створювати дублікат системи поряд для перевірки гіпотези може бути також неможливим через великі інвестиції. Тому у випадку з ЦД можна використати наступне копмпромісне рішення, яке в своїй основі було запропоновано у статті [45]. Цикл впровадження починається з аналізу вимог, а супроводжується не побудовою фізичного прототипу, а створення віртуальної моделі, її оцінки і лише після цього перехід до фізічної реалізації (рис. 11.7).

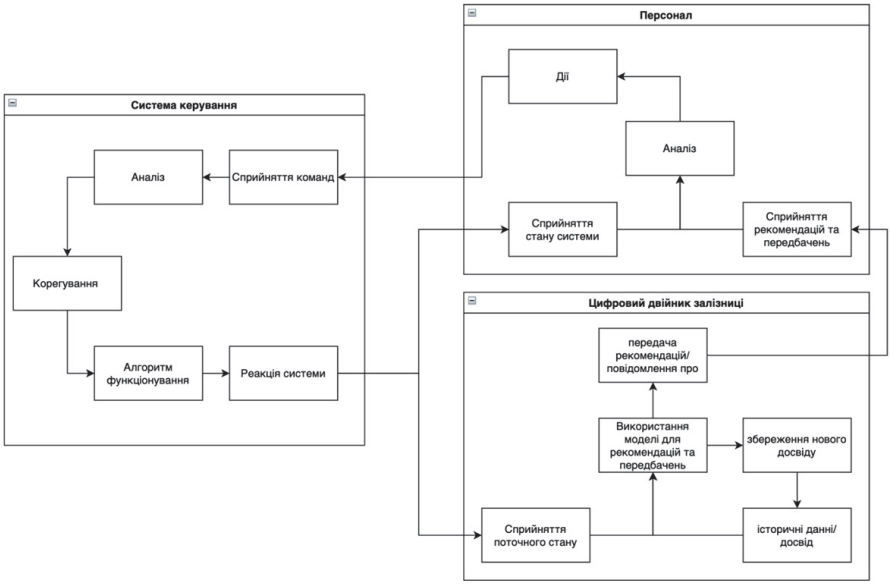



Рисунок 11.6 – Структурна схема процедур взаємодії систем мікропроцесорної централізації з ЦД та людиною

Головні переваги такого підходу полягають в наступному: порівняно низькі початкові інвестиції для перевірки гіпотези, можливість перевірки концепту і гіпотези раніше, немає необхідності втручатися у вже працюючі системи, можливість досить просто перетворювати моделі у фізичні системи вже після усіх верифікацій та перевірок. Тобто швидка і відносно недорога розробка моделі у віртуальній середі, яка може бути протестована

зацікавленими сторонами якомога раніше у циклі впровадження нового рішення, а також можливість робити ці перевірки віддалено. Крім того, при частковій наявності деяких фізичних компонентів системи на етапі моделювання, це все можна поєднати під однією віртуальною приборною панеллю для створення враження єдиної системи, що допоможе під час верифікацій, інтеграційного тестування та презентацій і перевірок зацікавленими сторонами [46].

Таким чином, запропонована схема системи мікропроцесорної централізації ЗА, а також процедура впровадження такої системи, може за відносно невеликі інвестиції і відносно невеликий проміжок часу допомогти перевірити концепт та побудувати доповнення до системи на базі ЦД, що буде виступати у ролі системи підтримки прийняття рішень для оперативного персоналу.

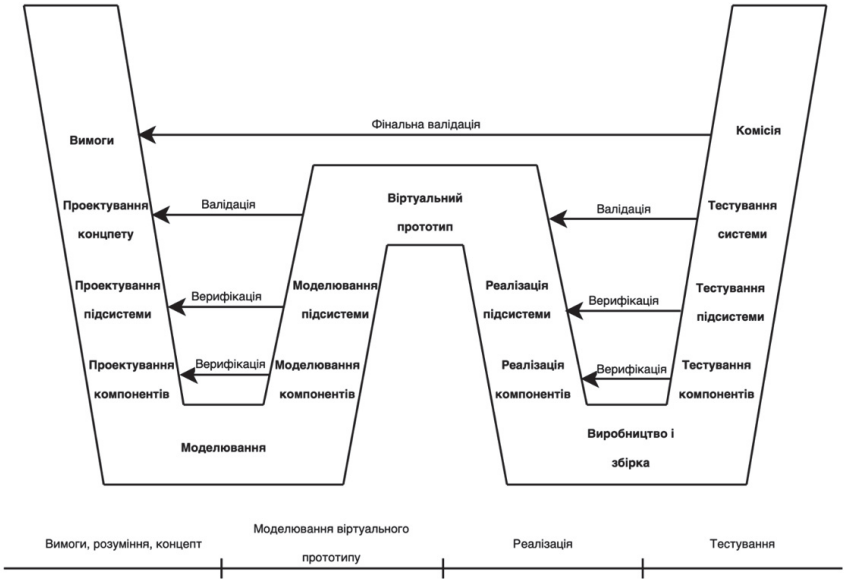



Рисунок 11.7 – Структура процедури впровадження нових рішень на базі ЦД

**Інструменти та підходи для можливої реалізація пілотного проекту.** На відміну від CAD/CAE моделювання, ЦД – це не лише модель, а й наявність фізичного двійника із постійним обміном даними для проведення аналізу і оптимізацій, прив'язаних до життєвого циклу об'єкту, із використанням нових знань від ЦД для data driven рішень (на основі даних) або для автоматичних команд у зворотньому зв'язку від ЦД до фізичного двійника. Тому ЦД - це

завжди мультидисциплінарне рішення, яке включає одразу декілька окремих напрямків у технологіях. Зазвичай типову архітектуру можна умовно поділити на декілька шарів:

1. Апаратне забезпечення - сенсори, пристрої, роутери та інші компоненти на рівні фізичного двійника. Тут обов'язково повині бути компоненти, які вміють підключатися до інтернету (edge gateway, iot хаби, крайові сервери, мікроконтролери та інші).
2. Транспорт та інтеграція - це рівень зв'язку пристроїв із сервером, передача даних, аутентифікація та інтеграція із рішеннями по обробці даних. Зазвичай тут використовується передача даних по протоколу HTTP або MQTT. Спочатку відправник аутентифікується із паролем або JWT токеном, після чого дані сприймаються сервером або хмарою (через REST API, MQTT брокери черги повідомлень).
3. Обробка даних – отримання, очищення, моделювання даних, верифікація, управління і т.д. В залежності від розміру системи чи даних та поставлених задач тут можна використовувати як прості веб сервіси написані на мовах Python, Scala, Java та інші, так і інструменти для великих даних - побудова конвеєрів даних із Spark, Kafka, Hadoop або аналогічних хмаринх інструментів та платформ - Google Dataflow, Databricks lakehouse, Amazon Kinesis, Azure Data Factory, Confluent.
4. Аналіз і моделювання – це рівень виконання передбачень, оптимізацій, аналізу, симуляцій та використання інших моделей. Це може бути як CAD або ML моделі, так і інструменти для побудови звітів, різні симуляції, що використовують отримані дані (звичайна база даних, або рішення як озеро даних чи сховище даних – data lake та data warehouse).
5. Рівень використання – в залежності від потреб це може бути або інструменти для візуалізації (Tableau, Power BI, Looker) для прийняття рішень на основі даних, або додатки (мобільні застосунки чи веб портали), або команди, що в автоматичному режимі відправляються як зворотній зв'язок до фізичних двійників (автономне дистанційне управління).

Такі рішення зазвичай потребують великої кількості інструментів та обчислювальних ресурсів, тому, як правило, використовують послуги хмарних постачальників та їх сервіси (AWS, GCP, Azure).

Також можна описати питання, які слід поставити, перед створенням рішення на базі ЦД [48]:

- Обрати рівень ЦД - чи це рівень компоненту, продукту, системи. Моделювання та створення ЦД може вимагати певних ресурсів. Тому краще спочатку сфокусуватися на конкретній частині продукту чи процесу і конкретних проблемах, які з точки зору економіки мають великий вплив на певні характеристики, які прагнуться бути покращенними із ЦД.
- Краще починати із одного аспекту або типу моделі - чи то покращенний моніторинг, чи передбачення відмов, моделювання певних сценаріїв, або допомога на етапі проектування. Після тестування та

впровадження одного із аспектів, можна розширювати двійник й на інші аспекти.

– Обрати тип моделювання - чи то модель фізична (наприклад на базі фундаментальних законів фізики/хімії, CAD моделі), або це буде моделювання на базі даних (тренування штучного інтелекту із використанням історичних даних).

– Оцінити можливі інвестиції - в обладнання для досягнення необхідного рівня цифровізації (сенсори, пристрої) та можливості працювати із великими потоками даних (мережеве обладнання, сервери для крайових обчисленнь); в хмарну інфраструктуру; в програмне забезпечення (інструменти для аналітики та симуляцій); в розробку, аналітику та галузевих експертів.

– Розглянути доцільність використання вже готових рішень або певних платформ, що вирішують частину задач. Як наприклад, IBM Digital Twin Exchange, Azure Digital Twins, GE Digital Twins, Oracle IoT Digital Twin Framework, Siemens Digital Twin software.

## 11.6. Особливості галузей

Відповідно до огляду та аналізу можна зробити висновок, що провідними галузями промисловості з точки зору дослідження та впровадження ЦД є: виробництво, розумне місто, охорона здоров'я, аерокосмічна та автомобільна промисловість.

Однак ЦД все активніше застосовуються в освіті, будівництві, залізничному транспорті та інших сферах. У таблиці 11.3 підсумовано основні характеристики та проблеми для кожного із розглянутих доменів.

Велика кількість підходів, порблем та рішень для ЦД є загальними одразу для кількох або навіть усіх галузей. Це тому, що існує багато проблем, які ортогональні до індустрій. Наприклад, ЦД як інструмент для навчання можна використовувати для підготовки автомобільних інженерів або щоб допомогти дослідникам вивчити та візуалізувати деякі процеси. Або проблема безпеки даних актуальна при будь-якому застосуванні ЦД.

Таблиця 11.3 – Особливості та проблеми розглянутих галузей ЦД

| № | Галузь | Особливості | Виклики |
|---|---|---|---|
| 1 | Виробництво | Прогнозне обслуговування та моніторинг виробничої лінії; моделювання та перевірка продукту перед виробництвом; підтримка операторів через ЦД обладнання та досвід AR/VR; ЦД як інструмент для інженерного навчання | Обробка великих даних; безпека даних; безпека ЦД; складність системи для багатопрофільних і багатоступінчастих виробничих процесів; висока вартість впровадження й розроблення системи |

Продовження табл. 11.3

| | | | |
|---|---|---|---|
| 2 | Автомобільна | Моделювати поведінку та стан автомобіля у різних середовищах та умовах; моніторинг поточного стану автомобіля та прогнозне обслуговування; покращений досвід продажів із ЦД та AR/VR; індивідуальні функції та виправлення | Оброблення та безпека великих даних; безпека автомобіля у разі перехоплення контролю над ЦД; комунікація – автомобіль може бути оффлайн |
| 3 | Медицина | ЦД людини; ЦД органів – вища точність діагностики та прогнозування лікування; прогностичне виявлення хвороби; моделювання біологічних процесів; ЦД для системи управління медичним центром і контролю обладнання | Етичні питання до ЦД людей; конфіденційність медичних даних і складність оброблення через цю конфіденційність; важко змоделювати або зрозуміти деякі біологічні процеси |
| 4 | Аерокосмічна галузь | Моделювання місій; авіамоделювання та тестування компонентів перед виробництвом; моніторинг стану повітряного судна; динамічне прийняття рішень за поточним станом апарату | Низька пропускна здатність і висока затримка для ЦД на землі для зв'язку з літаком; високі вимоги до безпеки – критична система |
| 5 | Розумні міста | Здатність досліджувати та моделювати соціальну діяльність для різних ситуацій; модель відкритого міста як інструмент демократії та управління; містобудування та моделювання аварій/катастроф | Розумне місто – гетерогенна система з великою кількістю інтеграцій; величезний обсяг даних та їх складність; конфіденційність громадян |
| 6 | Освіта | Покращена доступність; кожен учень може мати свою модель; здатність змоделювати ситуації, які важко знайти в реальності; безпека людей та обладнання | Проблеми з помилковим досвідом навчання (неточна модель), ЦД для навчання ручної діяльності та роботи в команді |

Продовження табл. 11.3

| | | | |
|---|---|---|---|
| 7 | Будівництво | Моделювання конструкції перед виробництвом; моніторинг на кожному етапі будівництва; розширена точність і нові моделі систем управління будівлею | Складні та багатоступінчасті процеси в будівельній галузі. Вимоги до безпеки та роботи у реальному часі проти точності – важко обробляти високоточні моделі будівель у режимі реального часу |
| 8 | Залізничний транспорт | Прогнозне обслуговування та безпека транспорту і залізнодорожніх вузлів або полотна. Моделювання трафіку та поведінки людей на станції для скорочення простою та покращення досвіду очікування пасажирів | Велика кількість та щільність критичних компонентів, що повинні бути цифровізовані для повного контролю |

Іншим аспектом, який виділяється в кожній галузі техніки, є застосування ЦД на різних рівнях протягом життєвого циклу розроблення продукту. Як під час створення медичного пристрою, так і під час будівництва нової будівлі, початковий прототип може бути протестований, а припущення підтверджене у віртуальному середовищі перед початком виробничого процесу. І двигуни літака, і виробнича машина вимагають технічного обслуговування, яке можна оптимізувати за допомогою ЦД і методів прогнозного технічного обслуговування.

**Переваги.** Основні переваги ЦД актуальні до усіх галузей:

– створення та тестування прототипів до реалізації, редизайн продукту;

– зниження ризиків иа витрат через відмови та аварії – прогнозне обслуговування, більш позорий моніторинг;

– безпека, доступність – через передбачення аварій або через можливість віртуального/віддаленого тестування/керування;

– зменшення відходів – може допомогти уникнути виробництва непотрібних продуктів, уникнення надмірного використання ресурсів;

– навчання – стає більш доступним, можливим, привабливим та безпечним.

**Виклики.** Проблеми в основному пов'язані зі складністю систем ЦД, точністю моделі та безпекою. Причиною цього є те, що системи ЦД

гетерогенні, мультидисциплінарні та включають широкий спектр складних технологій:

– складність – неоднорідні мультидисциплінарні системи включають занадто багато деталей, які важко зрозуміти і вимагають спеціалістів;

– відсутність стандартизації та інструментів – ЦД все ще активно розвиваються, досі немає чітко визначених протоколів, стандартів, фреймворків, які можуть допомогти побудувати такі системи. Наявні і проблеми у чіткості термінів;

– вартість – для використання повного потенціалу необхідно створити складну систему з великими моделями, залучити значні обчислювальні та людські ресурси. Якість ЦД залежить від якості віртуального представлення, і чим більше побудовано систем високої точності, тим більшу це має вартість.

– дані – ЦД потребують величезну кількість даних для моделей та передбачень, а також для історії і аудиту. Тому важливо добре організувати потоки даних, структури та якісноочищувати дані. Об'єднання даних вважається однією з поширених проблем ЦД;

– безпека та конфіденційність – оскільки велика кількість даних передається з різних джерел, важливо розробити безпечні з'єднання та механізми аутентифікації. Особливо це важливо для конфіденційних даних, таких як датчики медичних пристроїв, компоненти критичних систем, персональні дані;

– дані у реальному часі та точність – точніша модель ефективніше працює. Але такі точні моделі включають величезну кількість даних, які важко обробляти у реальному часі. Тому часто доводиться шукати компроміс.

Крім загальних викликів, кожна індустрія може мати свої унікальні [47]. Це або спецефічне застосування технології до індустрії - питання пропускної здатності зв'язку із космосом. Або якісь більш характерні до галузі питання - як етичні питання під час створення ЦД для людей. Інші приклади для деяких доменів вже було розглянуто у табл 1.3.

## 11.7. Висновки

ЦД все ще є досить новою концепцією, яка активно розвивається і має ряд невирішених проблем таких як загальні стандарти, визначенні терміни, фреймворки та прикладів для специфічних індустрій. В той же час існує широкий спектр успішних прикладів, де ЦД вже впроваджено та почало приносити переваги з точки зору безпеки, зниження витрат, прогнозованого обслуговування та прототипування. Самі цифрові двійники є логічним продовження розвитку кіберфізичних систем, IoT, великих даних та цифровізації бізнесів. Нова синергія технологій може суттєво змінити ефективність та безпеку виробництв, а також розширити можливості для нових розробок та досліджень.

Сучасні дослідження ЦД намагаються вирішити згадані проблеми, паралельно розширюючи зону їх застосування. Багато рішень активно обговорюються в рамках суміжних напрямків, таких як хмарні обчислення, IoT або аналітика великих даних. Але незважаючи на ці виклики, ЦД вважається однією з провідних технологій у космічному, військовому та виробничому секторах. Ринок ЦД та науковий інтерес до теми постійно зростає. Очікується, що ЦД будуть активно інтегровані в наше життя, не тільки в дослідженнях та виробництві, але й у звичайному житті: як громадяни на вулиці, як пацієнти в медичних центрах, як пасажири у транспорті, і як клієнти в торгових центрах або онлайн-магазинах.

Література


1. Jamwal A., Agrawal R., Sharma M., Giallanza A (2021), Industry 4.0 technologies for manufacturing sustainability: a systematic review and future research directions, Applied Sciences 2021, 11(12), 5725.
2. da Silva Mendonca R., de Oliveira Lins S., de Bessa I. V., de CarvalhoAyres F. A. Jr., de Medeiros R. L. P., de Lucena V. F. Jr (2022), Digital Twin Applications: A Survey of Recent Advances and Challenges, Processes 2022, 10, 744.
3. Concetta Semeraro, Mario Lezoche, Hervé Panetto, Michele Dassisti (2021), Digital twin paradigm: A systematic literature review, Computers in Industry, Elsevier, 2021, 130, pp.103469.
4. Why IoT is the Backbone for Digital Twin (2020), available at: https://www.ptc.com/en/blogs/corporate/iot-digital-twin (accessed August, 2022).
5. Top 10 Digital Twin Companies Impacting Industry 4.0 Innovations in 2021 (2022), available at: https://www.emergenresearch.com/blog/top-10-digital-twin-companies-impacting-industry-4-0-innovations-in-2021 (accessed August, 2022).
6. Singh M., Fuenmayor E., Hinchy E. P., Qiao Y., Murray N., Devine D (2021), Digital Twin: Origin to Future, Applied System Innovation 2021, 4, 36.
7. Kite-Powell J, Using Digital Twins And Drones To Capture Physical Environments (2021), available at: https://www.forbes.com/sites/jenniferhicks/2021/12/28/using-digital-twins-and-drones-to-capture-physical-enironments/?sh=31ca46e2556e (accessed August, 2022).
8. Asia/Pacific. Leads the Shift to Digital-First with 1 in 3 Companies Generating More Than 30% Revenues from Digital Products and Services By 2023 , IDC Predicts (2021), Available at: https://www.idc.com/getdoc.jsp?containerId=prAP48347921 (accessed August, 2022).
9. Dozortsev Victor, Digital twins in industry: genesis, composition, terminology, technologies, platforms, prospects. Part 2. Key technologies of digital

twins. Types of a physical object modeling (2020). Automation in Industry, 2020, No. 11, 3-10.

10. Qian C., Liu X., Ripley C., Qian M., Liang F., Yu W, Digital Twin – Cyber Replica of Physical Things: Architecture, Applications and Future Research Directions (2022), Future Internet 2022, 14, 64.

11. Fuller A., Fan Z., Day C., Barlow C, Digital Twin: Enabling Technologies, Challenges and Open Research (2020), IEEE Access 2020, 8, 108952–108971.

12. Singh M., Srivastava R., Fuenmayor E., Kuts V., Qiao Y., Murray N., Devine D (2022), Applications of Digital Twin across Industries: A Review, Appl. Sci. 2022, 12, 5727.

13. Botín-Sanabria D.M., Mihaita A.-S., Peimbert-García R.E., Ramírez-Moreno M.A., Ramírez-Mendoza R.A., Lozoya-Santos J.d.J, Digital Twin Technology Challenges and Applications: A Comprehensive Review (2022), Remote Sensing 2022, 14, 1335.

14. Autiosalo Juuso, Discovering the Digital Twin Web - From singular applications to a scalable network (2021), available at: https://aaltodoc.aalto.fi/handle/123456789/111416 (accessed August, 2022).

15. Shahriar M. (2020), Towards a Cyber-Physical Manufacturing Cloud through Operable Digital Twins and Virtual Production Lines, available at: https://scholarworks.uark.edu/etd/3739 (accessed August, 2022).

16. Kharchenko V., Morozova O., Illiashenko O., Sokolov S. A Digital Twin for the Logistics System of a Manufacturing Enterprise Using Industrial IoT. Information & Security: An International Journal 47, no. 1 (2020): 125–134. DOI:10.11610/isij.4708.

17. Stojanovic L., Uslander T., Volz F., Weibenbacher C., Muller J., Jacoby M., Bischoff T, Methodology and Tools for Digital Twin Management – The FA3ST Approach (2021), IoT 2021, 2, 717–740.

18. He F., Ong S. K., Nee A. Y. C, An Integrated Mobile Augmented Reality Digital Twin Monitoring System (2021), Computers 2021, 10, 99.

19. Kampczyk A., Dybeł K, The Fundamental Approach of the Digital Twin Application in Railway Turnouts with Innovative Monitoring of Weather Conditions (2021), Sensors 2021, 21, 5757.

20. Z. Lv D. Chen, M. S. Hossain, Traffic Safety Detection System by Digital Twins and Virtual Reality Technology (2022), IEEE International Instrumentation and Measurement Technology Conference (I2MTC), 2022, pp. 1-6.

21. Hazal Şimşek, Top 5 Use Cases of Digital Twin in Automotive Industry in 2022 (2022), available at: https://research.aimultiple.com/digital-twin-automotive/ (accessed August, 2022).

22. Hazal Şimşek, Best Digital Twin Applications & Use Cases in Healthcare in 2022 (2022), available at: https://research.aimultiple.com/digital-twin-healthcare/ (accessed August, 2022),

23. Y. Liu et al., A Novel Cloud-Based Framework for the Elderly Healthcare Services Using Digital Twin (2019), in IEEE Access, vol. 7, pp. 49088-49101, 2019.

24. Cory Kays & team (Aurora Flight Sciences), David Knezevic & Phuong Huynh (Akselos), Michael Kapteyn (MIT PhD student), Jacob Pretorius (Jessara Group), Development of a Predictive Digital Twin (2021), available at: https://kiwi.oden.utexas.edu/research/digital-twin (accessed August, 2022).

25. Shaun Waterman, Air Force Goes All in on Digital Twinning—for Bombs As Well As Planes (2021), available at: https://www.airforcemag.com/air-force-goes-all-in-on-digital-twinning-for-bombs-as-well-as-planes/ (accessed August, 2022).

26. Schrotter G., Hürzeler C, The Digital Twin of the City of Zurich for Urban Planning (2020), PFG 88, 99–112 (2020).

27. Tianhu Deng, Keren Zhang, Zuo-Jun (Max) Shen, A systematic review of a digital twin city: A new pattern of urban governance toward smart cities, Journal of Management Science and Engineering (2021), Volume 6, Issue 2, 2021, P.125-134.

28. David N. Ford, Charles M. Wolf, Smart Cities with Digital Twin Systems for Disaster Management (2020), Journal of Management in Engineering Vol. 36, Issue 4 (July 2020).

29. Shahat Ehab, Chang T. Hyun, Chunho Yeom, City Digital Twin Potentials: A Review and Research Agenda (2021), Sustainability 13, no. 6: 3386.

30. Joe David, DEVELOPMENT OF A DIGITAL TWIN OF A FLEXIBLE MANUFACTURING SYSTEM FOR ASSISTED LEARNING (2018), Available at: https://www.researchgate.net/publication/335234337_DEVELOPMENT_OF_A_DIGITAL_TWIN_OF_A_FLEXIBLE_MANUFACTURING_SYSTEM_FOR_ASSISTED_LEARNING (accessed August, 2022).

31. Christian Stary, Claudia Kaar, Sabrina Oppl, Dominik Schuhmann, Johannes Kepler University Linz, Tangibles and Digital Twins: Toward Meaningful Learning Support in CyberPhysical System Development (2022), ISBN: 978-1-912532-28-5.

32. Yitmen I., Alizadehsalehi S., Akıner ˙I., Akıner M. E, An Adapted Model of Cognitive Digital Twins for Building Lifecycle Management (2021), Applied Sciences 2021, 11, 4276.

33. Salem T., Dragomir M, Options for and Challenges of Employing Digital Twins in Construction Management (2022), Applied Sciences 2022, 12, 2928.

34. What is industry 4.0? (2016), available at: http://www.industriall-union.org/industry-40-the-industrial-revolution-happening-now/ (accessed August, 2022).

35. Aheleroff S., Xu X., Zhong R. Y., Lu Y, Digital Twin as a Service (DTaaS) in Industry 4.0: An Architecture Reference Model (2021), Advanced Engineering Informatics 2021, 47(2), 101225.

36. Padovano A., Longo F., Nicoletti L., Mirabelli G, A Digital Twin based Service Oriented Application for a 4.0 Knowledge Navigation in the Smart Factory (2018), IFAC-PapersOnLine 2018, 51(11), 631–636.

37. Al-Ali A. R., Gupta R., Batool T. Z., Landolsi T., Aloul F., Al Nabulsi A, Digital Twin Conceptual Model within the Context of Internet of Things (2020), Future Internet 2020, 12, 163.

38. Start Innovating with Digital Twins Technology, Available at: https://www.perforce.com/p/resources/vcs/digital-twins-technology (accessed August, 2022).

39. Hou Lei, Shaoze Wu, Guomin Zhang, Yongtao Tan, Xiangyu Wang, Literature Review of Digital Twins Applications in Construction Workforce Safety (2020), Applied Sciences 11, no. 1: 339.

40. Kaewunruen Sakdirat, Sresakoolchai Jessada, Lin Yi-hsuan, Digital twins for managing railway maintenance and resilience (2021), Open Research Europe. 1. 91. 10.12688/openreseurope.13806.1.

41. Dirnfeld Ruth, Digital Twins in Railways (2022), 10.13140/RG.2.2.32690.68804.

42. Kampczyk Arkadiusz, Dybeł Katarzyna, The Fundamental Approach of the Digital Twin Application in Railway Turnouts with Innovative Monitoring of Weather Conditions (2021), Sensors. 21(17). 5757. DOI:10.3390/s21175757.

43. Katharina Rombach, Towards a Data-driven Operational Digital Twin for Railway Wheels (2022) available at: https://youtu.be/5igWA9wuDdw (accessed August, 2022).

44. Гаєвський Віталій Вікторович, Удосконалення технічної експлуатації систем мікропроцесорної централізації на основі оперативної ідентифікації та локалізації порушень (2021)

45. Wagg D., Worden Keith, Barthorpe Robert, Gardner Paul, Digital Twins: State-of-The-Art Future Directions for Modelling and Simulation in Engineering Dynamics Applications (2020). ASCE-ASME J Risk and Uncert in Engrg Sys Part B Mech Engrg. 6. 10.1115/1.4046739.

46. Eric Lutters, Roy Damgrave, The development of Pilot Production Environments based on Digital Twins and Virtual Dashboards (2019), Procedia CIRP, Volume 84, Pages 94-99, ISSN 2212-8271, DOI:10.1016/j.procir.2019.04.228.

47. Sun, Y.; Fesenko, H.; Kharchenko, V.; Zhong, L.; Kliushnikov, I.; Illiashenko, O.; Morozova, O.; Sachenko, A. UAV and IoT-Based Systems for the Monitoring of Industrial Facilities Using Digital Twins: Methodology, Reliability Models, and Application. Sensors 2022, 22, 6444. DOI:10.3390/s22176444.

48. Altexsoft, Digital Twins: Components, Use Cases, and Implementation Tips (2021) available at: https://www.altexsoft.com/blog/digital-twins/

# 12. ТЕХНОЛОГІЇ ІНТЕРНЕТА РЕЧЕЙ ДЛЯ ПОБУДОВИ БЕЗПЕЧНИХ ВЕБ-ОРІЄНТОВАНИХ СИСТЕМ

**Є. В. Бабешко, Є. В. Мерзлікін**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 12.1. Вступ

Парадигма Інтернету речей (IoT) є ключовим кроком на шляху задуму та створення сучасних систем. Об'єднання інтелектуальних пристроїв і серверів, підключених до потенційної безлічі сенсорних і виконавчих вузлів, становить потужну інфраструктуру для розроблення додатків і систем, які підвищують інтелект і можливості користувачів-людей. Це охоплює від електронної охорони здоров'я до розумного виробництва та автоматизації, розумних міст та багато іншого.

Основною концепцією IoT є надшвидке наскрізне з'єднання між усіма пристроями, засноване на таких досягненнях у мережах, як технологія 5G і нова концепція 6G. Інтеграція з іншими парадигмами та технологіями ICT, такими як Cloud Computing, Fog Computing, DevOps, та програмні структури, надала набір інструментів для створення безпрецедентних додатків [1].

Сьогодні більшість систем IoT або інтенсивно використовують веб-взаємодії, або, принаймні, мають певну частину, яка використовує веб-протоколи та інструменти. Не тільки зручність використання цих систем і програм, але й зв'язок між пристроями базується на веб-технологіях і веб-протоколах завдяки стандартам W3C [2]. Веб-взаємодії покладаються на протоколи HTTP [3] і HTTPS [4] і схему REST [5]. Крім того, більшість платформ IoT побудовано на веб-інтерфейсі, який діє як централізована інформаційна панель і уніфікований сервер керування для зв'язку з пристроями системи.

Причина, чому безпека систем IoT дедалі більше порушується, залежить від їхньої природи. Вони складаються з великої кількості різнорідних пристроїв, таких як датчики, приводи, комп'ютерні вузли та сервери; вони сильно взаємопов'язані (зокрема, доступ до Інтернету) і покладаються на програмні платформи, здебільшого розроблені як випадкові системи. Крім того, впровадження веб-технологій розширило зону атаки для кіберзлочинців.

Існуючі підходи до підвищення безпеки в IoT зосереджені на підмножині компонентів, які містять ці системи, здебільшого пов'язаних із криптографією, мережевою передачею, маршрутизацією тощо. Сторона безпеки програмного забезпечення також була досліджена за допомогою методів аналізу коду, безпеки операційної системи, судової експертизи, і т. д.

Цього недостатньо в критичних підсистемах IoT, оскільки помилка може мати катастрофічні наслідки. Однак існує складна рівновага:

 з одного боку, критично важливі системи IoT повинні забезпечувати гарантії безпеки, щоб зменшити їх вразливість,

 з іншого, – серверам IoT може знадобитися інтенсивне використання веб-технологій, програмних інфраструктур і бібліотек. які дозволяють розробнику створити функціональні можливості, які інакше були б неможливі.

### 12.1.1. Мотивація

Різновидом IoT є промисловий (індустріальний) інтернет речей (Industrial Internet of Things, IIoT). IIoT є сукупністю мереж і пов'язаного з ними виробничого обладнання, доповненого програмним забезпеченням (ПЗ) і вбудованими датчиками.

Системи призначені для збору, обміну даними та можуть керуватися автоматично, без участі людини.

Індустріальний інтернет речей склався як загальна концепція застосування Інтернету речей (IoT) до промислового сектору. Ключовою технологією концепції Індустрія 4.0 вважається Інтернет речей, яка більше фокусується на ефективності промислових процесів (рис. 12.1).

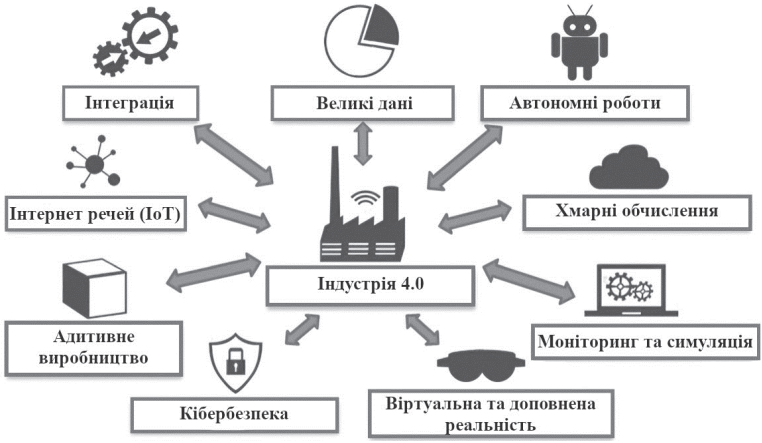



Рисунок 12.1 – Приклад індустріального Інтернету речей (IIoT) і технологій «Індустрія 4.0»

Бачення IIoT включає всі аспекти промислових операцій, зосереджуючись не тільки на ефективності процесів, але й на управлінні активами, обслуговуванні тощо. Важливими є такі тенденції:

 сучасна концепція Індустрія 4.0 поступово витісняє класичні ізольовані від мережі Інтернет системи управління від одного вендора;

- на заміну їм приходять індустріальні IoT системи – розподілені системи, у яких обмінюються даними різноманітні пристрої, активно використовуються хмарні та веб-технології;
- відкритість систем такого типу робить їх уразливими до кібератак;
- вплив атак може призвести до суттєвих фізичних та економічний збитків.

Індустріальний Інтернет речей значною частиною базується на застосуванні веб-технологій. Це зручно, оскільки дозволяє проводити моніторинг системи у реальному часі, вчасно впливати на неї, але і зростає потенційна загроза кібератак (рисунок 12.2).


Рисунок 12.2 – Ризики безпеки IIoT

Приклад атаки на IIoT систему: хакери зламали водоочисні споруди у Флориді, отримали доступ до внутрішньої платформи ICS і змінили рівень хімікатів, зробивши воду небезпечною для споживання (рисунок 12.3). Сталося це тому, що хакери отримали доступ до комп'ютерної системи співробітника. Його система надавала можливість віддалено усувати проблеми системи водоочисних споруд. Таким чином їм вдалося змінити кількість гідроксиду натрію у воді від 100 частин/мільйон до 11100 частин/мільйон.

Таким чином компанії будь-якого розміру та всіх галузей, щоб уникнути кіберкатастрофи (рисунок 12.4), повинні враховувати:

- використання технологій VPN: забезпечення безпечного тунелю і облікових даних, які надаються співробітникам для доступу до внутрішніх ресурсів і захисту критичних систем.
- правильна реєстрація та виключення: коли співробітники приєднуються до компанії та залишають її, важливо переконатися, що доступ надається лише за потреби та негайно скасовується, коли працівники залишають її.
- розділення доступу до мережі: гарантуйте, що працівники мають доступ лише до тих систем, які їм потрібні.

 важливо розміщувати різні системи в різних мережах, доступ до яких мають лише ті групи співробітників, яким вони потрібні, щоб гарантувати, що в разі злому менше систем може бути скомпрометовано.

 виділені робочі пристрої: у такі часи, як швидкий перехід до роботи з дому в 2020 році, коли багато співробітників отримували віддалений доступ до систем, надання співробітникам спеціального пристрою замість того, щоб дозволяти працівникам отримувати доступ до корпоративної мережі зі своїх власних пристроїв, дасть IT-відділам найбільший контроль над своєю інфраструктурою.

 постійне навчання співробітників: навчити співробітників розпізнавати фішингові листи так само важливо, як і запровадити захисні системи. У міру того як зловмисники знаходять нові способи проникнення в мережі, підготовка та оновлення співробітників лише посилить безпеку мережі.


Рисунок 12.3 – Хакер зламує водоочисні споруди Флориди, змінює рівень хімікатів [37]

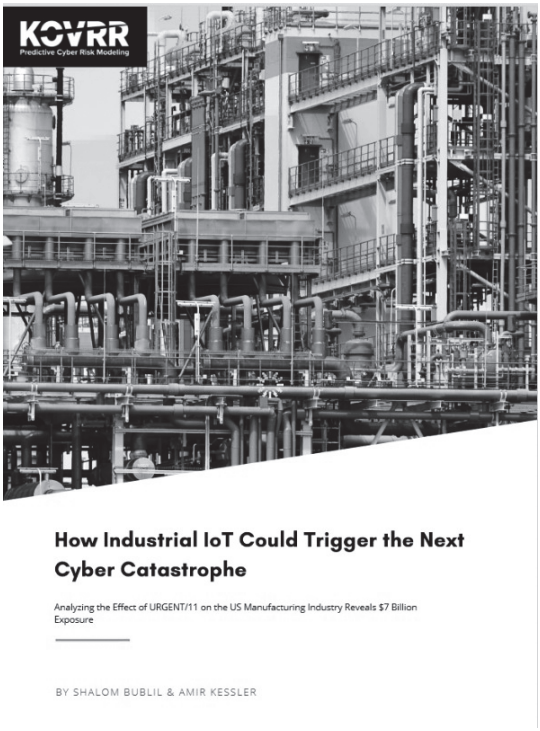



Рисунок 12.4 – Як індустріальний IoT може спровокувати наступну кіберкатастрофу [36]

12.1.2. Мета і структура

Метою даного розділу є аналіз існуючих методів, засобів та технологій організації веб-орієнтованих індустріальних IoT систем та проблеми забезпечення їх кібербезпеки.

У підрозділі 12.2 наведено класифікацію проаналізованих джерел. Підрозділ 12.3 містить результати аналізу джерел за виділеними напрямами класифікації. У підрозділі 12.4 наведено отримані результати. У підрозділі 12.5 сформульовано висновки та наступні кроки.

12.2. Класифікація джерел

Проаналізовані джерела було класифіковано за такими напрямами:

□ порівняння та аналіз огляду літератури: [19, 20, 21, 22, 23, 24, 25, 26, 27, 30, 33, 34]

- оцінювання безпеки інструментів веб-додатків: [1, 2, 3, 4, 5, 7, 9, 16, 17, 34, 35, 38, 39, 40]
- методи та рішення забезпечення безпеки Інтернету речей (IoT): [6, 7, 8, 9, 10, 11, 12, 13, 14]
- Структура систем індустріального Інтернету речей (IIoT) і технологій «Індустрія 4.0»: [9, 15, 16]

Крім того, джерела було класифіковано за типом (рисунок 12.5).


Рисунок 12.5 – Класифікація джерел

12.3. Аналіз джерел за напрямами

12.3.1. Методи оцінювання та забезпечення кібербезпеки веб-орієнтованих індустріальних IoT систем на різних етапах життєвого циклу

Основні організації, такі як Міжнародна організація стандартизації (ISO) 27001, перерахували вимоги щодо впровадження та покращення параметрів безпеки для веб-додатків [38].

Для компаній і розробників регулярно публікується десятка найкритичніших загроз безпеці [39], щоб підвищити поінформованість про безпеку веб-додатків та зменшити збитки, які завдають потенційні вразливості додатків. Однак, відповідно, існують рекомендовані шаблони програмування, які допомагають користувачам виправляти небезпечні конфігурації та мінімізувати ризики [40].

Нижче наведено вибраний набір ризиків безпеки для веб-інтерфейсів:

Порушений контроль доступу. Контроль доступу змушує користувачів діяти суворо в межах області програми, до якої їм надано дозвіл. Якщо застосована політика контролю доступу не є успішною для даного користувача, він/вона матиме доступ до цінних ресурсів, які повинні бути недоступні для нього/його.

Криптографічні збої призводять до розкриття конфіденційних даних. З цієї причини шифрування даних має відповідати найновішим стандартам,

уникаючи того, щоб системи (браузер, база даних тощо) використовували застарілі криптографічні функції для шифрування даних, паролів тощо. Також необхідно класифікувати рівень безпеки всіх системних даних щоб визначити, які криптографічні алгоритми використовувати та як сертифікувати та підтвердити приймач.

Ін'єкція включає низку ризиків, наприклад, міжсайтовий сценарій (XSS) і ін'єкцію SQL. Ін'єкція відбувається, коли введені користувачем дані не перевіряються належним чином і не фільтруються, і вони стають частиною веб-сторінки або бази даних; якщо зловмисний вхід безпосередньо використовується програмою, ворожі дані можуть бути впроваджені в записи програми або базу даних, спричиняючи довготривалу шкоду системі. Це можна частково виправити, відокремивши команди від даних і перевіривши вхідні дані за допомогою відповідних фільтрів; за допомогою безпечніших API та елементів керування SQL, які обмежують неавторизовані операції та уникають розголошення конфіденційних даних; і переглядаючи вихідний код.

Неправильна конфігурація безпеки часто виникає під час увімкнення, вимкнення, встановлення чи видалення функцій; залишити обліковий запис і пароль за умовчанням без змін; використання застарілих комплектуючих; і не ввімкнути останні функції безпеки.

Помилки ідентифікації та автентифікації зазвичай є результатом атак грубої сили та/або криптографічних збоїв за допомогою сценаріїв, які постійно намагаються автоматично перевірити комбінації імен користувачів і паролів. Такі дії відомі як атаки грубою силою. Такі ризики існують, коли існує база даних облікових даних із слабкою безпечною автентифікацією (наприклад, користувач адміністратора з паролем abc123); або коли зловмисники користуються витоком ідентифікаторів сесії, щоб отримати контроль над зв'язком. Більше кроків для перевірки (наприклад, код, безпечні запитання, використання інших пристроїв, таких як мобільний, розпізнавання обличчя тощо) можуть зменшити ризик атак грубою силою. Іншими заходами захисту є використання складних паролів і видалення простих тестових облікових записів.

Якщо програма залежить від ненадійних бібліотек, модулів та/або інших ресурсів, можуть виникнути збої програмного забезпечення та цілісності даних.

Збої в журналі безпеки та моніторингу мають бути постійним завданням для виявлення активних атак. Створення журналів невдалих спроб входу є основною стратегією запобігання цьому; однак цього може бути недостатньо з кількох причин: деякі невдалі спроби входу можуть не реєструватися; крім того, попередження та помилки можуть не давати чіткого уявлення про ситуацію. Розробник повинен переконатися, що: усе керування доступом і збої можуть бути записані з достатньою кількістю даних користувача, зареєстровані дані зашифровані в разі ін'єкції та встановлено механізм звіту про помилки.

Підроблення запиту на стороні сервера відбувається, коли програма отримує віддалений ресурс без перевірки URL-адреси, поданої користувачем. У результаті зловмисник може надіслати створений запит на несподівані

посилання, навіть якщо додаток захищено брандмауерам або VPN. Прикладна сторона має уникати цього, фільтруючи та перевіряючи всі дані, надіслані користувачем, встановлюючи ряд схем URL-адрес і уникаючи надсилання необроблених відповідей клієнту.

За статистикою, 19% уразливостей сканованих веб-додатків дозволяють зловмиснику контролювати програму та операційну систему. У 2019 році звіт про вразливості в корпоративних інформаційних системах [1] показав, що майже 75% векторів проникнення в локальну мережу (LAN) мають уразливості в захисті веб-додатків. Подібним чином уразливості в процесі розроблення створюють серйозні загрози для веб-додатків.

Крім того, змін у налаштуваннях конфігурації достатньо для усунення лише 17% вразливостей. Більшість із них мають низький рівень тяжкості. П'ятдесят відсотків витоків спричиняють розкриття облікових даних облікового запису та витік персональних даних, а близько 91% відсканованих веб-додатків зберігають і обробляють особисті дані.

Вибір правильної методології тестування на проникнення для конкретного типу вразливості [43] відіграє важливу роль у скануванні веб-додатку для виявлення вразливості. Більше того, автоматизація тестування веб-додатків корисна для pen-tester, що не тільки зменшило трудову роботу, час, ресурси та вартість, але й зменшило залежність тестувальників від знань людини [44].

Крім того, сканер зберіг людські знання о pen-testing, створивши виконувані комп'ютерні програми. Таким чином, розроблення автоматизованих сканерів безпеки веб-додатків зробила ручне тестування популярною тенденцією досліджень. У цій сфері розробники перетворюють методи pen-testing (Penetration test/Тест на проникнення) веб-додатків у виконувані програми, щоб краще виявляти вразливості веб-додатків.

### 12.3.2. Огляд літератури щодо оцінювання веб-додатків

Оцінка була зосереджена головним чином на можливостях оцінки вибраних веб-додатків щодо XSS (міжсайтових сценаріїв), впровадження SQL, впровадження коду та несправних елементів керування доступом. Виявлені вразливості були класифіковані, щоб уможливити процес оцінки сімнадцяти різних вразливостей.

У цьому огляді літератури обговорюються різні інструменти. Незважаючи на те, що порівняння та підсумковий аналіз літератури представлено в таблиці 1, деякі з цих досліджень обмежуються кількома виявленими вразливими місцями, а інші лише порівнюють обмежені сканери.

Однак у цьому опитуванні 11 власних сканерів веб-додатків із відкритим вихідним кодом порівнювалися з можливостями виявлення 10 найпопулярніших уразливостей OWASP.

Таблиця 12.1 – Аналіз авторів і напрямів досліджень

| Дослідник | Оцінений сканер | Оцінені вразливості | Основні внески | Обмеження |
|---|---|---|---|---|
| Doupé et al. (2010) [19] | Acunetix 174, AppScan Burp, Grendel-Scan, Hailstorm, Milescan, N-Stalker, NTOSpider, Paros, W3af, Webinspect | Cross-Site Scripting, SQL Injection, Code, Injection, Broken Access Controls | точність, час виконання та показники загрози | Обсяг виявлення вразливостей дуже обмежений |
| Bau et al. (2010) [20] | Acunetix, Cenzic, McAfee SECURE, WebInspect, N-Stalker, QualysGuard | Cross-Site Scripting, SQL Injection, Cross Channel Scripting, Session Management, Cross-Site Request Forgery | Scanner footprint, виявлення вразливості, помилкове спрацьовування | Сканери можуть виявляти прості атаки ін'єкцій XSS і SQL, але не змогли виявити форму ін'єкцій XSS і SQL другого порядку |
| Parvez et al. (2015) [21] | Acunetix, Rational AppScan, ZAP | SQL Injection, Stored XSS | Швидкість виявлення для SQLI та XSS | Порівнюються лише два комерційні сканери |
| Suteva et al. (2012) [22] | NetSparker, N-Stalker, OWASP-ZAP, W3af, Iron WASP, Vega | XSS, SQL Injection, Command Injection, File Inclusion | NetSparker має кращий рівень виявлення, ніж інші | Linux, MAC, Windows |
| El drissi et al. (2017) [23] | BurpSuite, Acunetix, Netsparker, AppSpider, Arachni, Wapiti, SkipFish, W3AF, IronWASP, ZAP and Vega | XSS, SQL Injection, Local and Remote File Inclusion, Path Traversal | Уразливості XSS і SQL мають вищий рівень виявлення, Arachni краще працює в інструментах з відкритим кодом | Linux, MAC, Windows |

| | | | | |
|---|---|---|---|---|
| Elahen, Claire et al. (2013) [24] | User Generated Content (UGC) evaluation | Assessing and Ranking UGC Assessment of healthcare systems | Хибнопозитивний рівень | |
| Mohit et al. (2017) [25] | BurpSuite, Acunetix, Wapiti, SkipFish, Netsparker, W3AF, AppSpider, Arachni, ZAP, Vega | Cross site scripting, SQL injection, Remote code execution, File inclusion | Точність, відкликання та F-measure | Обмежується ризиками помилкової тривоги та точністю |
| Gaurav et al. (2018) [26] | Penetration system for malware detection and web assessment using cloud services | OWASP's Vulnerabilities | Економічне рішення для веб-оцінки | Обмежується окремими веб-додатками |
| Mehreen et al. (2018) [27] | | | оцінка на основі опитувальника з позитивними результатами | Для цілей оцінки технічні деталі не надаються |
| Ashikali et al. (2018) [28] | W3af, Havij, Fimap, Metasploit, Acunetix, Nexpose | OWASP'S Vulnerabilities | Основні методи оцінки вразливості та тестування на проникнення | Не проведено жодного порівняльного аналізу |
| Rawaa (2016) [38] | Paros, Wapiti, Skipfish, Nikto, Wfuzz, NetSparker, HP WebInspect | SQL Injection, Cross Site Scripting | Аналітичне порівняння шести інструментів з відкритим кодом | Обговорюється обмежена кількість уразливостей |
| Mark and Rudolph (2006) [30] | Commercial and Free/Open-source Tools | | Переваги та недоліки окремих інструментів | Окремі засоби не користуються популярністю |

Продовження табл. 12.1

| | | | | |
|---|---|---|---|---|
| Fang et al. (2018) [31] | AppScan, AWVS, Netspark, Vega, W3af | | Виділена функція за допомогою згорткової нейронної мережі та точно ідентифікує сканери | Модель не можна динамічно покращувати |
| Alsaleh, Mansour et al. (2017) [32] | Arachni, Wapiti, Skipfish | SQL, Cross Site Scripting | Сканер Arachni може перевірити 100 відсотків тестів SQL | Немає істотної різниці в продуктивності між вибраними сканерами |
| Terry et al. (2018) [33] | OpenVAS, Kismet, Aircrack, SQLMAP, Wapiti | Buffer overflow, Cross Site Scripting | | Обмежено лише інформаційною системою охорони здоров'я |

Інструменти оцінювання веб-додатків можна знайти як з відкритим кодом, так і запатентовані. Запатентовані інструменти зазвичай пропонують безкоштовні пробні пакети для користувачів і тестерів пера; однак їх можливості та функції обмежені. Існує кілька аспектів, пов'язаних із вибором одного сканера над іншим.

Сканер повинен:

- Підтримка протоколів і алгоритмів автентифікації, які використовуються веб-додатками.
- Підтримувати основні типи методів доставки вхідних даних і мати можливість виявляти вразливості у веб-додатку з низьким рівнем хибно-позитивних результатів.
- Перебувати в межах технічних можливостей особи, яка буде ним користуватися.
- Бути стабільним та регулярно оновлюватись останніми оновленнями безпеки, щоб покрити поточні вразливості, які виявляються.
- Бути обраним, зберігаючи при цьому вартість та ліцензування у межі бюджету.

12.3.2.1. Існуючі методи, засоби та технології організації веб-орієнтованих індустріальних IoT систем та проблеми забезпечення їх кібербезпеки

Було проаналізовано 15 статей за 2019 – 2022 роки. Проведений аналіз показав, що для досліджуваних систем пропонуються такі рішення:

- методи забезпечення безпеки IoT за допомогою Машинного (Machine Learning) та Глибокого навчання (Deep Learning) [10, 11]
- метод оцінювання кібербезпеки веб-додатків на основі систем керування вмістом [10, 41]
- метод забезпечення кібербезпеки веб-систем шляхом вибору заходів захисту [42]
- методи збору та аналізу даних пристроїв IoT [42]
- методи побудови VPN-лацюгів між кінцевими користувачами віртуальної мережі [42].

12.3.2.2. Огляд методів машинного та глибокого навчання для безпеки Інтернету речей (IoT)

Рішення:

- ML і DL дозволив розробити різні потужні аналітичні методи, які можна використовувати для підвищення безпеки
- аналіз трафіку на основі потоку дозволяє виявляти зловмисну поведінку без необхідності поглибленого аналізу пакетів
- інтеграція ML і DL з блокчейном для безпеки IoT

Проблеми та виклики (рисунок 12.6):

- такий підхід вимагає певного часу для поглибленого аналізу, і він погано масштабується
- цей метод вимагає досить багато часу на навчання ML

12.3.2.3. Поглиблене вивчення та виявлення вразливостей і атак на веб-додатки: систематичний огляд

Архітектура веб-додатків (рисунок 12.7):

- веб-програми є основним мережевим рішенням для надання стандартних веб-служб.
- розроблення цих додатків базується на клієнтському та серверному розробленні.
- серверна сторона включає веб-сервер, веб-додаток і сервер бази даних; він використовує серверні мови сценаріїв, включаючи .NET, PHP та інші. Клієнтська сторона працює у веб-браузері користувача за допомогою зовнішніх мов сценаріїв, зокрема CSS/HTML, Javascript тощо.

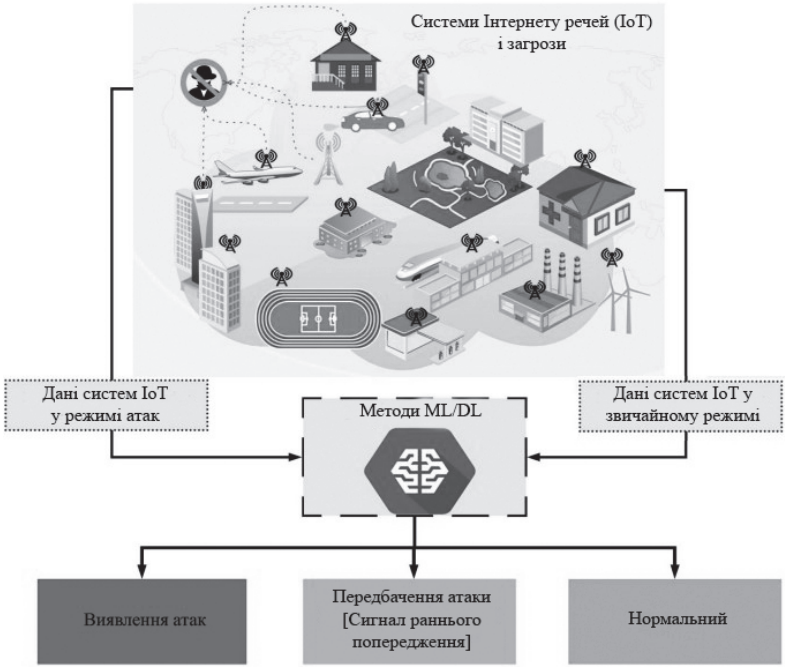



Рисунок 12.6 – Ілюстрація потенційної ролі ML/DL у безпеці IoT

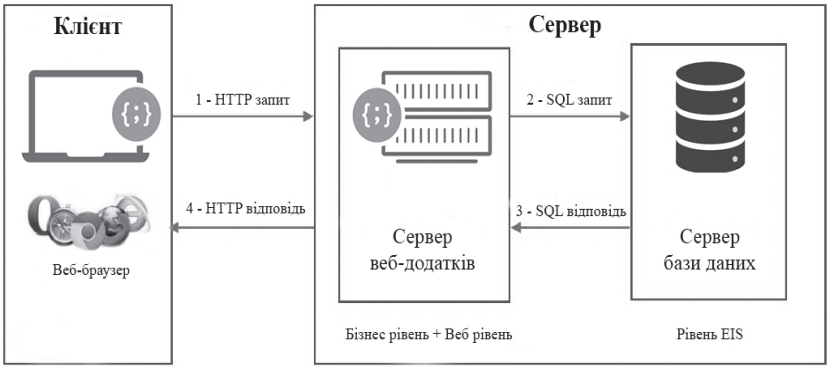
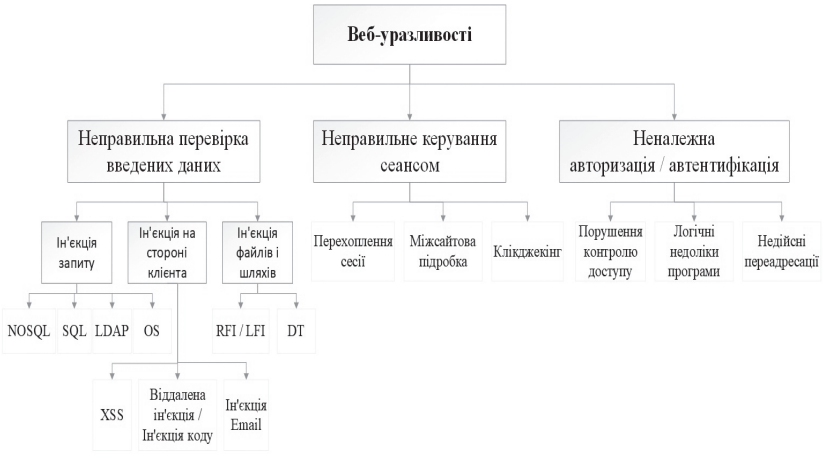



Рисунок 12.7 – Огляд веб-архітектури

Вразливості веб-додатків (рисунок 12.8):

- міжсайтове підроблення запитів (CSRF)

- ☐ впровадження SQL і міжсайтовий сценарій (XSS) є прикладами веб-атак
- ☐ підроблення сертифікатів
- ☐ DDOS-атаки
- ☐ слабкі паролі адміністраторів
- ☐ використовування ненадійних пристроїв




**LDAP** (Lightweight Directory Access Protocol) - Полегшений протокол доступу до директорій / каталогів
**OS** (Operating Systems) - Операційна система
**RFI** (Remote File Inclusion) - Віддалене виконання коду
**LFI** (Local File Inclusion) - Локальне виконання коду, в межах сервера
**DT** (Directory Traversal attack) - Атака обходу каталогу використовує недостатню перевірку безпеки

Рисунок 12.8 – Типи веб-уразливостей [41]

12.3.2.4. Огляд атак, уразливостей та засобів захисту в Індустрії 4.0 з новими викликами в галузі суверенітету даних

Зі збільшенням кількості пристроїв, підключених до мереж з підтримкою Industry 4.0, поверхня атаки також розширюється. Останні впровадження Industry 4.0 включають такі технології, як хмарні обчислення, штучний інтелект, пристрої CPS або IoT.

У разі зламу ці пристрої можуть завдати серйозної шкоди матеріальним благам, наприклад продуктам на виробничій лінії, або нематеріальним благам, таким як витік конфіденційної інформації чи промислових секретів.

Представлено загальний систематичний огляд поточних атак на кібербезпеку, уразливостей і засобів захисту в сценаріях Індустрії 4.0 і 5.0.

Представлено детальний аналіз і класифікацію щодо атак, уразливостей і захисту окремих досліджень.

12.4. Результати аналізу

У проаналізованих публікаціях акцентуються проблеми кібербезпеки IoT-систем в цілому та індустріальних IoT-систем, зокрема:

- безпека мережевого рівня архітектури IoT залишається привабливою до атак;
- існують ризики безпеки для веб-інтерфейсів;
- криптографічні збої також трапляються зі зростанням IoT;
- пропонуються локальні рішення, спрямовані на усунення виявлених уразливостей у конкретних компонентах.
- відсутній методологічний підхід забезпечення кібербезпеки вказаних систем на різних етапах життєвого циклу;
- розроблені методи та підходи швидкого виявлення уразливостей безпеки працюють не достатньо ефективно;
- сканери безпеки веб-застосунків мають різні слабкі сторони і часто генерують неправильні результати тестування.

12.5. Висновки

12.5.1. Обговорення результатів

Оцінка веб-додатків містить багато дій, спрямованих на підвищення загальної безпеки та надійності проти різних кібератак. Розробники та тестувальники використовують різні інструменти для сканування додатків веб-серверів і динамічного виявлення всіх можливих уразливостей. Багато сканерів веб-уразливостей потребують вдосконалення для мінімізації рівня хибнопозитивного виявлення. Хибнопозитивні уразливості здебільшого з'являються з високою частотою за допомогою автоматизованих інструментів, що може призвести до неправильної оцінки безпеки цільових веб-систем.

Впровадження заходів безпеки, таких як шифрування, автентифікація, контроль доступу, безпека мережі та додатків для пристроїв Інтернету речей та їх властивих уразливостей є не повністю ефективним.

Існує низка методів, підходів виявлення уразливостей у веб-орієнтованих індустріальних IoT системах, які покращують всю систему загалом, але потребують удосконалення.

12.5.2. Наступні кроки

Подальшим напрямом роботи є аналіз існуючої нормативної бази в галузі Інтернету речей, IIoT, їх кібербезпеки з врахуванням веб-складової з метою формування профілю і перевірки виконання вимог до систем такого типу.

Література

1. García-Valls, M. Introducing the new paradigm of Social Dispersed Computing: Applications, Technologies and Challenges. [Text] / M. García-Valls, A. Dubey, V. Botti // Journal of Systems Architecture. – 2018. – No. 91. – P. 83-102. doi: https://doi.org/10.1016/j.sysarc.2018.05.007.

2. W3C. The World Wide Web Consortium [Electronic resource] // The World Wide Web Consortium – 2021. – URL: www.w3.org. – 30.05.2022.

3. Fielding, R. HyperText Transfer Protocol v1.1; HTTP (RFC 2616). [Text] / R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee // The Internet Society: Reston, VA, USA. – 1999.

4. Rescorla, E. HTTP over TLS – RFC 1818. [Text] / E. Rescorla // Internet Engineering Task Force. – 2000.

5. Fielding, R.T. Representational State Transfer (REST). Architectural Styles and the Design of Network-based Software Architectures. [Text] / R. T. Fielding // Thesis, University of California, Irvine, CA, USA. – 2000. – Vol. 5. – P. 76-147.

6. Pedreira, V. A Review of Attacks, Vulnerabilities, and Defenses in Industry 4.0 with New Challenges on Data Sovereignty Ahead, Sensors. [Text] / V. Pedreira, D. Barros, P. Pinto // MDPI journals, Sensors. – 2021. – Vol. 21(15), No. 5189. doi: https://doi.org/10.3390/s21155189.

7. García-Valls, M. Improving Security of Web Servers in Critical IoT Systems through Self-Monitoring of Vulnerabilities. [Text] / L. Song, M. García-Valls // MDPI journals, Sensors. – 2022. – Vol. 22, No. 5004. doi: https://doi.org/10.3390/s22135004.

8. Fang, Z. A model checking-based security analysis framework for IoT systems. [Text] / Z. Fang, H. Fu, T. Gu, Z. Qian, T. Jaeger, P. Hu, P. Mohapatra // Journal of High-Confidence Computing. – 2021. – No. 100004. doi: https://doi.org/10.1016/j.hcc.2021.100004.

9. Sarwar, A. Enhanced Anomaly Detection System for IoT Based on Improved Dynamic SBPSO. [Text] / A. Sarwar, A. Alnajim, S.N.K. Marwat, S. Ahmed, S. Alyahya, W.U. Khan // MDPI journals, Sensors. – 2022. – Vol. 22, No. 4926. doi: https://doi.org/10.3390/s22134926.

10. Beyzanur C. Overview of Cyber Security in the Industry 4.0 Era. [Text] / B. Ervural, B. Ervural // Managing The Digital Transformation. – 2017. – P. 267–284. doi: https://doi.org/10.1007/978-3-319-57870-5_16.

11. Alaoui, R.L. Deep Learning for Vulnerability and Attack Detection on Web Applications: A Systematic Literature Review. [Text] / R.L. Alaoui, E.H. Nfaoui // MDPI journals, Future Internet. – 2022. – Vol. 14, No. 118. doi: https://doi.org/10.3390/fi14040118.

12. Al-garadi, M. A. A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security. [Text] / M. A. Al-garadi, A. Mohamed, A. Al-Ali, M. Guizani // IEEE Internet of Things Journal. – 2020. – No. 19890478. doi: https://doi.org/10.1109/COMST.2020.2988293.

13. Shahid, J. A Comparative Study of Web Application Security Parameters: Current Trends and Future Directions. [Text] / J. Shahid, M.K. Hameed, I.T. Javed, K.N. Qureshi, M. Ali, N. Crespi // MDPI journals, Applied Sciences. – 2022. – Vol. 12, No. 4077. doi: https://doi.org/10.3390/app12084077.

14. Pathak, G. LPWAN Key Exchange: A Centralised Lightweight Approach. [Text] / G. Pathak, J. Gutierrez, A. Ghobakhlou, S.U. Rehman // MDPI journals, Sensors. – 2022. – Vol. 22, No. 5065. doi: https://doi.org/10.3390/s22135065.

15. Surej, H. I. A FeedForward–Convolutional Neural Network to Detect Low-Rate DoS in IoT. [Text] / H. I. Surej, M. Ma, R. Su // Engineering Applications of Artificial Intelligence. – 2022. – Vol. 114. No. 105059. doi: https://doi.org/10.1016/j.engappai.2022.105059.

16. Ferrer, B. R. Connecting Web-Based IoT Devices to a CloudBased Manufacturing Platform. [Text] / B. R. Ferrer, W. M. Mohammed, E. Chen, J. L. Martinez Lastra // IEEE Internet of Things Journal. – 2017. – No. 17431808. doi: https://doi.org/10.1109/IECON.2017.8217516.

17. Aazam, M. Deploying Fog Computing in Industrial Internet of Things and Industry 4.0. [Text] / M. Aazam, S. Zeadally, K. A. Harras // IEEE Internet of Things Journal. – 2018. – No. 18133157. doi: https://doi.org/10.1109/TII.2018.2855198.

18. Kabla, H. Applicability of Intrusion Detection System on Ethereum Attacks: A Comprehensive Review. [Text] / H. Kabla, M. Anbar, S. Manickam, T. A. Al-Amiedy, P. B. Cruspe, A. K. Al-Ani, S. Karuppayah // IEEE Access Journal. – 2022. – Vol. 10, No. 21863800. doi: https://doi.org/10.1109/ACCESS.2022.3188637.

19. Gupta, A. The IoT Hacker's Handbook. [Text] / A. Gupta // Apress Berkeley, CA. – 2019. doi: https://doi.org/10.1007/978-1-4842-4300-8.

20. Doupé, A. Why Johnny can't pentest: An analysis of black-box web vulnerability scanners. [Text] / A. Doupé, M. Cova, G. Vigna // In Proceedings of the International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, Bonn, Germany. – 2010. – Springer: Berlin/Heidelberg, Germany. – 2010. P. 111–131. doi: https://doi.org/10.1007/978-3-642-14215-4_7

21. Bau, J. State of the art: Automated black-box web application vulnerability testing. [Text] / J. Bau, E. Bursztein, D. Gupta, J. Mitchell // IEEE Symposium on Security and Privacy, Berleley/Oakland, CA, USA. – 2010. – IEEE: Piscataway, NJ, USA. – 2010. – P. 332–345. doi: https://doi.org/10.1109/SP.2010.27.

22. Parvez, M. Analysis of effectiveness of black-box web application scanners in detection of stored SQL injection and stored XSS vulnerabilities. [Text] / M. Parvez, P. Zavarsky, N. Khoury // IEEE: Piscataway, NJ, USA. – 2015. – P. 186–191. doi: https://doi.org/10.1109/ICITST.2015.7412085.

23. Suteva, N. Evaluation and testing of several free/open source web vulnerability scanners. [Text] / N. Suteva, D. Zlatkovski, A. Mileva // Conference for Informatics and Information Technology (CIIT 2013), Bitola, Macedonia. – 2013.

24. Idrissi, S. Performance evaluation of web application security scanners for prevention and protection against vulnerabilities. [Text] / S. Idrissi, N. Berbiche,

F. Guerouate, M. Shibi // International Journal of Applied Engineering Research. – 2017. – Vol. 12, No. 21. – P. 11068–11076.

25. Momeni, E. A survey on assessment and ranking methodologies for user-generated content on the web. [Text] / E. Momeni, C. Cardie, N. Diakopoulos // ACM Comput. Surv. (CSUR). – 2016. – Vol. 48, No. 41. doi: https://doi.org/10.1145/2811282.

26. Kumar, M. An Efficient Model for Web Vulnerabilities Detection based on Probabilistic Classification. [Electronic resource] / M. Kumar, S. Majithia, S. Bhushan // Int. J. Technol. Comput. (IJTC). Techlive Solut. – 2016. – P. 50–58. – URL: www.semanticscholar.org/paper/An-Efficient-Model-for-Web-Vulnerabilities-based-on-Kumar-Majithia/f09ddc0501358e234a5f8e9ebec359beb91db8f1. – 12.04.2022.

27. Raj, G. Security testing for monitoring web service using Cloud. [Text] / G. Raj, M. Mahajan, D. Singh // IEEE: Piscataway, NJ, USA. – 2018. – No. 18043392. – P. 316–321. doi: https://doi.org/10.1109/ICACCE.2018.8441734.

28. Ahmed, M. Web application prototype: State-of-art survey evaluation. [Text] / M. Ahmed, M. Adil, S. Latif // IEEE: Piscataway, NJ, USA. – 2015. – No. 15756740. – P. 19–24. doi: https://doi.org/10.1109/NSEC.2015.7396339.

29. Hasan, A. Web Application Safety by Penetration Testing. [Electronic resource] / A. Hasan, D. Meva // Int. J. Adv. Stud. Sci. Res. – 2018. – URL: www.academia.edu/38248493/Web_Application_Safety_by_Penetration_Testing. 12.04.2022.

30. Mohammed, R. Assessment of Web Scanner Tools. [Text] / R. Mohammed // Int. J. Comput. Appl. – 2016. – Vol. 133, No. 5. doi: https://doi.org/10.5120/ijca2016907794.

31. Curphey, M. Web application security assessment tools. [Text] / M. Curphey, R. Arawo // IEEE Secur. Priv. – 2006. – Vol. 4. – P. 32–41. doi: https://doi.org/10.1109/MSP.2006.108

32. Fang, Y. DarkHunter: A fingerprint recognition model for web automated scanners based on CNN. [Text] / Y. Fang, X. Long, L. Liu, C. Huang // In Proceedings of the 2nd International Conference on Cryptography, Security and Privacy, Guiyang, China. – 2018. – ACM: New York, NY, USA. – 2018. – P. 10-15. doi: https://doi.org/10.1145/3199478.3199504.

33. Alsaleh, M. Performance-based comparative assessment of open source web vulnerability scanners. [Electronic resource] / M. Alsaleh, N. Alomar, M. Alshreef, A. Alarifi, A. Al-Salman // Secur. Commun. Netw. – 2017. – URL: www.hindawi.com/journals/scn/2017/6158107. – 12.04. 2022.

34. Terry, M. A comprehensive security assessment toolkit for healthcare systems. [Text] / M. Terry, O.D. Oigiagbe // Colon. Acad. Alliance Undergrad. Res. J. – 2015. – Vol. 4. – P. 1–6.

35. Furrer, F. J. Safety and Security of Cyber-Physical Systems. [Text] / F. J. Furrer // Engineering dependable Software using Principle-based Development. – 2022. doi: https://doi.org/10.1007/978-3-658-37182-1.

36. Wu, D. Cybersecurity for digital manufacturing. [Text] / D. Wu, A. Ren, W. Zhang, F. Fan, P. Liu, X. Fu, J. Terpenny // J. Manuf. Syst. – 2018. – Vol. 48. – P. 3-12. doi: https://doi.org/10.1016/j.jmsy.2018.03.006.

37. Bublil, S. How Industrial IoT could Trigger the Next Cyber Catastrophe. [Electronic resource] / S. Bublil, A. Kessler // Kovrr. – 2020. – URL: www.kovrr.com/reports/how-industrial-iot-could-trigger-the-next-cyber-catastrophe-2.

38. Henriquez, M. Hacker breaks into Florida water treatment facility, changes chemical levels. [Electronic resource] / M. Henriquez // Security Magazine. – 2021. – URL: https://www.securitymagazine.com/articles/94552-hacker-breaks-into-florida-water-treatment-facility-changes-chemical-levels.

39. ISO/IEC 27001. – Information Technology. – Security Techniques. – Information Security Management Systems – Requirements. ISO/IEC International Standards Organization: Geneva, Switzerland, 2005.

40. Top 10 Web Application Security Ristsk. [Electronic resource] // The OWASP Foundation. – 2022. – URL: https://www.owasp.org.

41. Agreindra Helmiawan, M. Analysis of Web Security Using Open Web Application Security Project 10. [Text] / M. A. Helmiawan, E. Firmansyah, I. Fadil, Y. Sofivan, F. Mahardika, A. Guntara // International Conference on Cyber and IT Service Management (CITSM), Pangkal, Indonesia. – 2020. – P. 1–5. doi: https://doi.org/10.1109/CITSM50537.2020.9268856.

42. Application Security Verification Standard. [Electronic resource] // OWASP Application Security Verification Standard. – 2022. – URL: http://www.owasp.org/index.php/ASVS.

43. Morozova, O. Metody ta tekhnolohiyi zabezpe-chennya kiberbezpeky industrial'nykh i veb-oriyentovanykh system i merezh. [Text] / O. I. Morozo-va, A. O. Nicheporuk, A. H. Tets'kyy, V. M. Tkachov // Kharkivs'kyy aviatsiynyy instytut: Naukova robota. – 2021.

44. Bhorkar, G. Security Analysis of an Operations Support System. [Electronic resource] / G. Bhorkar // Aalto University. School of Science. Master's Programme in Computer, Communication and Information Sciences. – 2017. – URL: https://aaltodoc.aalto.fi/handle/123456789/29252.

**45.** Seng, L. K. The approaches to quantify web application security scanners quality: A review. [Text] / L. K. Seng, N. Ithnin, S. Z. M. Said // Int. J. Adv. Comput. Res. – 2018. – Vol. 8. – P. 285–312. doi: https://doi.org/10.19101/IJACR.2018.838012.

# 13. МЕТОДИ І ТЕХНОЛОГІЇ ПОБУДОВИ ТА ДИСТАНЦІЙНОЇ РЕКОНФІГУРАЦІЇ ВУЗЛІВ ВБУДОВАНОЇ СИСТЕМИ

**О. О. Вдовіченко, А. Є. Перепелицин**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 13.1. Вступ

Сучасна промисловість надає багато апаратних рішень для побудови систем на базі спеціалізованих мікросхем або мікропроцесорних систем (МПС) серійного виробництва [1]. Такі рішення знаходять своє застосування в бортових системах, включаючи аерокосмічні системи, де діють особливі вимоги до показників надійності та можливості реконфігурації.

Системи на базі мікропроцесорів містять програмний код, який вирішує поставлену задачу. Застосування подібних підходів дає можливість знизити ціну в розрахунку на один екземпляр системи, а використання МПС дозволяє гнучко модифікувати алгоритм у прошивці шляхом її перепрограмування.

Слід зазначити, що виробники готових рішень використовують далеко не всі можливості сучасної елементної бази для підвищення їх відмовостійкості або енергозбереження. Використання цих додаткових можливостей бази стає у нагоді при розробленні бортових систем і частин інфраструктури будівель.

Отже існує можливість забезпечення гнучкої зміни конфігурації технічних систем на базі МПС, спрямовані на врахування можливих відмов [2] або виправлення помилок конкретного екземпляру обладнання, що буде сприяти підвищенню його живучості у разі деградаційних відмов. Зазвичай апаратна надлишковість має доповнюватись програмною надлишковістю шляхом використання флеш-пам'яті.

Повсюдне застосування компонентів для швидкого прототипування вже дозволяє знизити трудовитрати і вартість перевірки працездатності окремого концепту цифрової системи [3]. Універсальність і доступність таких компонентів на основі мікроконтролерів дає можливість здійснювати таке прототипування дистанційно групою розробників [3].

Масове застосування адитивних технологій і 3d-друку дозволяє використовувати мініатюрні серійні компоненти на основі мікроконтролерів для побудови цифрових пристроїв, периферії та компонентів розумного будинку. Подальша мініатюризація та серійний випуск компонентів на основі мікроконтролерів дозволить безпосередньо додавати такі системи як витратний матеріал для 3d друку.

При суттєвому збільшенні кількості вузлів та зниженні вартості за один мікроконтролер стає актуальною задача діагностики стану вузлів, перепрограмування окремих вузлів, виявлення та відновлення деградаційних відмов [4, 5].

**Метою даної роботи** є підвищення надійності систем, побудованих із використанням апаратного забезпечення на базі мікроконтролерів. Для досягнення поставленої мети в даній роботі розглядаються та вирішуються такі **задачі**:

– відшукання методу комунікації набору мікроконтролерів у системі з можливістю уточнення вимог;

– відшукання можливості дистанційного перепрограмування окремих мікроконтролерів у складі цільової системи;

– пошук способу додавання надмірності в схему підключення мікроконтролерів для можливості реконфігурації;

– сформулювати елементи методу і послідовність розроблення систем, які передбачають можливість дистанційної діагностики, перепрограмування і реконфігурації їх окремих вузлів;

– розроблення прототипу МПС на основі розглянутого рішення конкретного виробника для демонстрації запропонованого підходу з метою внесення елементів надлишковості.

## 13.2. Аналіз можливостей швидкого проєктування вбудованої системи

Ефективним способом забезпечення зв'язку та енергоживлення для набору периферійних модулів системи виступають провідні лінії. Ці лінії можуть бути використані як для діагностики так і для перепрограмування окремих модулів. Саме тому виникає потреба в універсальному методі дистанційної комунікації, діагностики та програмування за універсальними лініями. Слід розглянути вже існуючі методи, конструкції та пристрої для виконання набору операцій, що розглядаються при побудові рішень з однотипних блоків. Більшість таких модулів мають єдиний послідовний інтерфейс.

В той самий час, використання роботизованих рішень значною мірою знижує трудомісткість і людський фактор. Однак, використання вузькоспеціалізованих дорогих рішень потребує великих матеріальних витрат на обладнання та підвищення кваліфікації працівників для роботи з ним, що робить їх економічно маловигідним.

### 13.2.1. Порівняльний аналіз компактних роботизованих систем

Повторне використання існуючих компонентів на етапі проєктування комерційних завдань є найбільш доцільним, оскільки дозволяє знизити трудовитрати скоротити час отримання пробної реалізації. Визначення властивостей існуючих компонентів вимагає проведення аналізу популярних компонентів роботизованих систем.

Серед існуючих автоматичних рішень можна виділити такі проекти, як роботичний черв’як під назвою "Вермі бот", який був розроблений групою

юних інженерів, вихованців центру любителів робототехніки "Солярис". Його основним призначенням є проведення аварійно-рятувальних робіт під землею: у канавах, каналізаційних тунелях та печерах. Також, розробники вважають доцільним використання робота для прокладальних робіт. Головними перевагами робота вважають: легкість виробництва та можливість пересування в усіх напрямах.

Основою цього рішення виступає модуль - самостійна одиниця пристрою, з сукупності яких він складається. Модульність пристрою дозволяє розширювати та модифікувати пристрій за потреби. Пристрій містить у собі модулі наступних типів: руховий модуль та модуль керування. Руховий модуль зібраний з пластикових хрестоподібних конструкцій, що рухаються електромотором. Під час роботи конструкція змінює свої геометричні параметри, чим і приводить пристрій до руху. Модуль керування містить у собі джерело живлення та плату керування. Плата контролює подачу струму на двигуни та визначає порядок роботи модулів руху.

Кінцевою метою розробки автори поставили вдосконалення пристрою для використання у космічній галузі під час колонізації планет [6].

Ще одним, вартим уваги, рішенням можна вважати серію швидких роботів-черв'яків "Inchworm", розроблених медичною лабораторією Ізраїльського Технічного Інституту, який являють собою серію компактних пристроїв на поршневому ходу. Їх основним призначенням є обслуговування систем маленьких труб та судин у медичній апаратурі.

Пристрій складається з модифікованого електромотора та крокових кілець. Мотор оснащений валом зі спеціальним різьбленням, яке задає траєкторію вертикального переміщення кільця під час руху. На "крокових кільцях" із зовнішньої сторони знаходяться малоковзні ніжки для фіксації пристрою в трубці. З внутрішньої сторони кільця розташовується стрижень, призначений для розміщення у різьбленні на валу.

У подальшому планується їх модифікація для роботи в серцево-судинних системах людини. [7].

Наостанок, з не черв'якоподібних можна виділити робота-павука BionicWheelBot від компанії "Festo". Прямого призначення, на момент дослідження, робот не має і розроблявся компанією з метою демонстрації її можливостей Робот являє собою збільшену у 21,6 разів роботичну копію існуючого павука виду Cappараceae aureola.

Має у своїй конструкції тіло, та 4 пари кінцівок. Здатний пересуватися двома способами: за допомогою перестановки кінцівок та перекатами, попередньо склавши кінцівки у формі колеса [8].

Порівняння основних характеристик розглянутих прикладів компактних роботизованих систем наведено в таблиці 13.1.

Таблиця 13.1 – Відмінні риси пристроїв компактних роботизованих систем

| | Швидкість, м/с | Довжина, м | Вага, г |
|---|---|---|---|
| Вермі бот | 0,1 | 1,3 | 1400 |
| Inchworm | 0,05 | 0,06 | 40 |
| BionicWheelBot | 0,5 | 0,55 | 580 |

Проведений аналіз показує, що сфера застосування системи вимагає від пристрою чіпкості, малих габаритів та маневреності. З розглянутих вище аналогів можна назвати:

– модульність системи, що дозволяє розширювати систему залежно від завдань;

– наявність сервоприводних кінцівок для адаптації до середовища, в якому пристрій буде працювати і можливість колективізації завдання шляхом використання більш ніж одного екземпляра системи.

13.2.2. Аналіз технологічної бази для швидкої побудови модульних рішень

Існують майданчики, набори модулів та конструктори які дозволяють швидко і при мінімальних витратах сил і засобів створювати технічні рішення для вузькоспеціалізованих завдань.

Серед багатьох можна виділити такі рішення:

1) IQBX – електромеханічний конструктор із пустотілих блоків, а також модулів, які можуть підключатися до них.

До базових блоків набору відносять: пустотілі конструкційні (одно-секційні та багатосекційні), акумуляторні, блоки з електронікою та блоки-приводи з вбудованим у середину мотором-редуктором.

Крім базових блоків набір містить у собі кріплення, Lego-сумісні перехідники та шлейфи проводів для з'єднання електричних та електромеханічних частин між собою [9];

2) Lego Technic – лінійка сумісних пластикових стрижнів, електричних модулів і деталей механічного призначення Lego.

Комплекти часто містять у собі такі види деталей: балки та пластини для каркасу, шестерні та осьові штифти, пневматичні елементи та сервоприводи, блоки живлення [10].

3) Arduino – набір апаратно-програмних засобів з відкритим кодом для побудови простих систем автоматики та робототехніки.

Набір являє собою широку лінійку радіоелектронних компонентів, мікроконтролерів різних потужностей і форм, готових пристроїв з уніфікованими інтерфейсами та методами кріплення [11].

Компоненти саме цього набору стали найпопулярнішими та суттєво спростили процес прототипування систем на основі мікроконтролерів.

Для описаних вище наборів можна провести порівняльну характеристику, представлену в таблиці 13.2.

Виходячи з перерахованого вище можна зробити висновок, що доступні технологічні рішення можуть забезпечити достатню елементну базу для покриття більшості технологічних завдань. Залежно від вимог до систем, що розробляються, набори можуть комбінуватися між собою з метою поповнення недоліків один одного.

Таблиця 13.2 – Основні переваги наборів інструментів

| **Назва технології** | **Переваги** | **Недоліки** |
|---|---|---|
| IQBX | – наявність міцних елементів для конструкції<br>– велике різноманіття виконавчих модулів<br>– наявність стандартизації модулів<br>сумісність з іншими платформами | – висока вартість компонентів<br>– відсутність градації у розмірах компонентів<br>– обмежений тираж та важкодоступність на ринку<br>– складність під час побудови |
| Lego Technic | – висока доступність компонентів у всьому світі<br>– велике різноманіття виконавчих модулів<br>– простота під час складання | – висока вартість компонентів<br>– крихкість компонентів<br>– низька сумісність з іншими платформами |
| Arduino | – низька вартість компонентів<br>– висока доступність компонентів у всьому світі<br>– спрощена мова програмування<br>– інтуїтивно зрозуміле розширюване середовище розробки<br>– наявність готових рішень для датчиків та виконавчих механізмів<br>– велика кількість рішень у відкритому доступі(популярно у спільноті) | – відсутність оптимізації у стандартних бібліотеках<br>– неможливість використання усіх ресурсів мікроконтроллеру при використанні бібліотек |

## 13.3. Аналіз варіантів для забезпечення реконфігуропридатності вузлів вбудованої системи

У рамках даної роботи під вузлом або компонентом системи слід розуміти окремий мікроконтролер. Система може складатися з одного або декількох вузлів, що взаємодіють. Ця взаємодія здійснюється завдяки підключенню вузлів до єдиного каналу зв'язку і є дистанційною по відношенню до вузла.

Під діагностикою вузлів системи розуміється процес передачі інформації про їх технічний стан, у тому числі з опціональною можливістю опитування та перевірки ліній на схемі для дослідження стану монтажних з'єднань у схемі підключення мікроконтролера.

Під перепрограмуванням вузла системи мається на увазі процес заміни програми у флеш-пам'яті конкретного мікроконтролера без безпосереднього підключення до нього програматорів або зовнішніх інструментів.

Під реконфігурацією вузла системи розуміється процес зміни основної програми в мікроконтролері для реалізації нових функцій або підтримки частини існуючих функцій у разі виявлення відмови.

### 13.3.1. Забезпечення можливості реконфігурації модульних систем на етапі проєктування

У більшості технічних рішень важливим є показник надійності. Найчастіше, базові її показники досягаються шляхом підбору якісних компонентів та впровадженням запобіжних елементів. Однак, у особливо важливих системах вводиться комплекс резервів для життєво необхідних вузлів. У штатному режимі резерв може бути задіяний повністю (гарячий резерв), завдяки чому частково знижується загальне навантаження на систему. Також резерв може бути задіяний частково (теплий резерв). У такому разі частина передбачених функцій модуля залишається незадіяною.

Розглянуті плати та компоненти, зважаючи на свою поширеність, являють собою фабричні рішення. Це означає, що процес їх виробництва та попередньої побудови стандартизовані. У свою чергу, стандартизація забезпечує високу базову надійність за рахунок якісних комплектуючих. Так само, для універсальності та високої сумісності, у багатьох компонентів вводиться надмірність портів, елементів схем та ліній живлення. Така надмірність дозволяє підвищити ресурс пристрою на етапі проектування.

Крім відомих способів забезпечення надійності пропонується використовувати можливість реконфігурації технічних систем як засіб часткового відновлення функціональності у разі їх відмови. Для цього передбачається виділення сегментів системи та їх градація за ступенем важливості та угруповання за схожістю елементної бази. У такому випадку з'являється можливість використовувати елементи менш важливих сегментів системи як резерв для сегментів з вищим пріоритетом.

Також не виключена ручна модифікація подібних блоків з метою розширення їхнього потенціалу як компонент системи. Прикладом може бути плата розширення для модуля конвертера UART-USB, що реалізує додаткові органи керування передавачем [12].

### 13.3.2. Аналіз можливих варіантів забезпечення реконфігуропридатності

**Побудова системи з використанням модулів із модифікованим завантажувачем** дозволяє суттєво спростити процес програмування і перепрограмування, а також тонко керувати параметрами чіпу.

Для більшості МП рішень є можливість завантажувати основний текст програми у чіп через bootloader, змінюючи існуючу програму. При цьому оригінальний bootloader може бути встановлений ще на етапі виробництва. Такий підхід дає можливість виключити використання спеціалізованого програматора, який є складовою системи та перепрограмувати МП, використовуючи лише лінію UART [13].

В цьому випадку актуальною є задача надання можливості одному чіпу у складі ансамблю подібних до нього чіпів перепрограмувати інші. Це робить можливим процес відновлення окремих вузлів, у складі системи, якщо під час розроблення системи така можливість була передбачена заздалегідь.

Якщо окремі вузли виступають в якості посередника для перепрограмування інших, стає можливим здійснення поетапного відновлення окремих компонентів системи без безпосереднього доступу до них. Окрім того одні компоненти МПС можуть виступати у якості діагностуючих. Знання топології плат і підключеної периферії дають можливість виконувати діагностику з'єднань на друкованій платі за рахунок завантаження в такі вузли модифікованих прошивок. Цей процес може здійснюватися за участю інших вузлів-посередників.

**Використання спеціалізованих програмних пакетів** для створення та модифікації прошивки (firmware) може суттєво спростити процес їх створення і забезпечити можливість параметризації за інформацією про конфігурацію і властивості МПС. Ці параметри можна доповнити даними на основі аналітичних результатів. Також у процесі розробки можуть бути використані елементи штучного інтелекту. Застосування всіх цих інструментів в комплексі дає можливість на основі специфікацій на розроблення системи передбачати можливу обв'язку, структуру для заданих елементів, а також передбачити можливості її подальшої реконфігурації.

Якщо технічна можливість перепрограмування була передбачена на етапі проектування, то окремі компоненти системи отримують можливість виконувати завантаження нових програм у склад інших для діагностики оточуючих схем і перевірки коректності підключення і монтажу. У випадку наявності доступу до firmware конкретного вузла системи існує можливість подальшого дистанційного програмування через вузли-посередники для відновлення оригінальних функцій кожного з них.

Значного спрощення процесу створення програмних пакетів можна досягти за допомогою використання гнучких інструментів програмування. Для апаратних рішень важливою є наявність підтримки усіх апаратних платформ у середовищах для їх програмування. У випадку відсутності необхідного пакету підтримки платформи, суттєвою перевагою середовища розробки може виступати наявність функції їх додавання. Залежно від постачальника програмного забезпечення для розробки, ця функція може бути реалізована як у вигляді окремої програми, так і у вигляді вбудованого інструментального засобу.

**Дистанційна діагностика** схем може бути реалізована навіть якщо спочатку вона не була передбачена. Завдання дистанційної діагностики пристрою може бути вирішене завдяки організації таких зв'язків комунікації та можливості реконфігурації за допомогою перепрограмування. Така можливість надає більш серйозні функції, ніж заздалегідь передбачені можливості діагностики на рівні кристалу одноразово запрограмованого мікроконтролера [14]. Перевага такого підходу ґрунтується на тому, що може бути використано весь вільний ресурс флеш-пам'яті чіпу мікроконтролера винятково для задач діагностики.

Таке рішення може дозволити виконати децентралізований процес перевірки працездатності системи в цілому. Це може бути здійснено за допомогою періодичної та планової перевірки її стану з поверненням компонентів до нормального стану у після цього.

Можливість застосування такого концепту для критичних систем має бути розглянуто окремо. Але запропонована ідея побудови перепрограмованих рішень такого роду заздалегідь має широкий спектр застосування в системах, де може бути виконана планова зупинка функціонування для діагностики з використанням таких розширених функцій.

**Відновлення самої схеми** у разі відмови однієї з ліній портів або механічного пошкодження схем також стає можливим при використанні перепрограмування всередині системи. Запропонований підхід дає можливість виконувати часткову реконфігурацію схем і повну зміну програми роботи складових частин апаратної системи за рахунок їх дистанційного перепрограмування безпосередньо всередині системи.

Часткова реконфігурація схеми може бути забезпечена модифікацією набору ліній у складі схеми окремого мікроконтролера зі заздалегідь передбаченою надмірністю. Така надмірність у випадку апаратної відмови ліній у складі мікроконтролеру, монтажу у складі плати чи інших несправностей поза чіпом може дозволити виконати ряд дій для повного або часткового відновлення функцій системи. Відновлення можливе там, де реалізована спроможність заміни монтажних з'єднань за допомогою резисторів підтяжки, ємностних зв'язків чи додаткових ліній комунікації.

**Використання лінії скидання reset в якості звичайного порту** загального призначення як додаткова можливість надається більшістю мікроконтролерів. Це може бути зроблено за допомогою модифікації ф'юзу з

використанням спеціалізованого програма тора [15]. Після виконання такої процедури програмування мікроконтролера за допомогою звичайного низьковольтного програматора стає неможливим і він втрачаємможливість скидання без модифікації ф'юзу. В свою чергу модифікація ф'юзу стає неможливою без високовольтного паралельного програматора, що ускладнює процес програмування мікроконтролера штатними засобами. Bootloader дає можливість використовувати штатний програматор без використання високовольтного паралельного програматора.

Якщо перед зміною ф'юза завантажити в мікроконтролер bootloader з підтримкою роботи без лінії reset, то можливість програмування такого компонента зберігається. Використання у складі bootloader спеціальної логіки та наявність у прошивці алгоритмів для емуляції сигналу скидання reset надає можливість ініціювати початок завантаження програми у мікроконтролер.

У випадку емуляції сигналу скидання reset можна об'єднати використання окремих ліній у складі мікроконтролера як для виконання функцій порту відповідно до вимог проекту користувача, так і для керування процесом завантаження програмного коду у мікроконтролер. Такий підхід дає можливість зробити процес завантаження нових програм у мікроконтролер більш гнучким оскільки дозволяє зберегти сумісність запропонованого підходу з існуючими засобами розроблення та програмування мікроконтролерів.

Перевагою запропонованого підходу є можливість звільнення однієї з ліній у складі мікроконтролера для використання у якості звичайного порту. Ця лінія може використовуватись в якості лінії керування периферійними пристроями, приєднаними до поточного мікроконтролера. Вона також може бути використана в якості резервного каналу взаємодії та обміну даними з іншими мікроконтролерами.

**Використання однієї лінії зв'язку** для підключення набору вузлів дозволяє спростити процедуру взаємодії та знизити структурну складність схеми. Це доречно на вирішення поодиноких завдань вибіркового перепрограмування і діагностики.

До популярних стандартів комунікації з та без використання диференціальних ліній і підключенням як польова шина слід віднести RS485, Profibus, CAN, Modbus [16], CC-Link, FlexRay, HART та інші. Також слід звернути увагу на зовсім новий стандарт 802.3cg-2019. Це 10-мегабітний Ethernet з єдиною парою ліній та прямим монтажним підключенням до багатьох пристроїв на великій відстані [17]. Недоліком цих стандартів комунікації є необхідність підтримки складних протоколів.

Відомі спеціалізовані однопровідні протоколи комунікації, серед яких слід зазначити 1-ware. Але найбільший інтерес з погляду універсальності представляє набір протоколів, сумісних з UART. Слід згадати GBus (з бібліотекою GyverBus) и M-Bus. Ця шина за протоколом сумісна з UART, і при цьому є однією двонаправленою лінією. Передавання по цій лінії може ініціювати будь-який з вузлів. Один починає передавати, а решта в цей момент приймає (рис. 13.1).

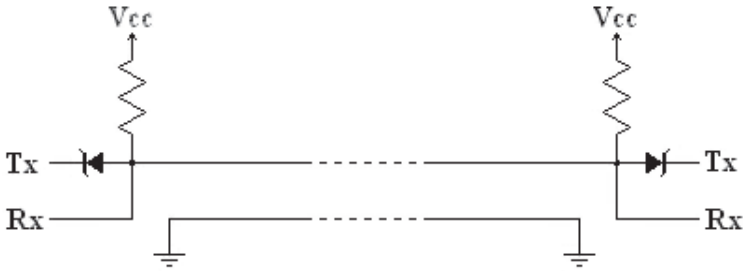



Рис.унок 13.1 – Приклад з'єднання 2 вузлів по одній лінії зв'язку

Більшість часу усі вузли такої шини перебувають у режимі прийому. У разі необхідності передачі, вузол запускає таймер і дивиться, чи лінія переведена в нуль. Якщо – не була, вузол починає передавати за допомогою звичайного асинхронного протоколу UART. Передача починається з нульового старт-біту, за яким слідують інформативні біти і стоп-біт.

Такий спосіб взаємодії дає можливість використовувати лише одну лінію з підтяжкою для підключення великої кількості мікроконтролерів. Резистор підтяжки потрібен для організації можливості організації взаємодії «більшість до більшості» у випадку інтенсивних обмінів. Також може бути доданий діод для можливості використання апаратної реалізації UART. Простий приклад використання такої лінії зв'язку представлений на рисунку 1.

Для реалізації розглянутого підходу не потребується складних апаратних рішень. Зберігається сумісність зі стандартними TTL контролерами та стандартними засобами моніторингу порту для інтерфейсу UART. Складністю такого підходу є необхідність доопрацювання програмної реалізації інтерфейсу UART. В частині мікроконтролерів існують апаратні рішення для обміну по інтерфейсу UART, які дозволяють виконувати прийом та передавання даних. Недоліком також є можливість апаратної відмови одного з мікроконтролерів та константної підтяжки лінії до нуля.

Перевагою такого підходу є простий варіант монтажу і проста топологія зв’язків. Це не вимагає організації високочастотних ліній як у I2C. Частоти обміну відповідають стандартному ряду UART.

**Боротьба з колізіями** є невід'ємною частиною під час обмінів між окремими мікроконтролерами по одній лінії зв'язку, яке передбачає можливість появи колізій.

Таким чином, для боротьби з колізіями під час обміну по одній лінії зв'язку пропонуються кроки:

1) перевірка контрольної суми і довжини посилки UART;

2) на основі таймера виділяються слоти в часі;

3) якщо шина зайнята або два вузли намагаються передати одночасно, то обидва переходять в режим очікування на певний випадковий для кожного вузла час, після закінчення якого обидва знов починають передавання;

4) кожен вузол має змогу контролювати передавання інформації, яка приходить йому на вхід.

**Перепрограмування окремих мікроконтролерів** у складі системи з використанням однієї лінії зв'язку є наступним кроком проектування систем, які передбачають реконфігурацію.

При реалізації такого підходу можуть використовуватись стандартні засоби розроблення та програмування мікроконтролерів із використанням однієї лінії. Це включає можливість створення та завантаження у мікроконтролер програм-прошивок для окремих вузлів, які зможуть перепрограмовувати інші вузли. Цей процес перепрограмування відбувається із використання спільної лінії зв'язку. Всі інші вузли в цей момент призупиняють обмін через цю лінію.

Існуючі засоби розроблення для мікроконтролерів і їх програмування надають можливість персоналізації і додаткових налаштувань.

В якості прикладу може бути розглянутий набір сумісних засобів розробки для мікроконтролерів AVR від компанії Atmel (з 2016 у складі MicroChip). Вони дають можливість виконувати перепрограмування через лінії reset, яка може бути підтягнута до 0, після чого завантажений у чіп bootloader починає свою роботу. Використання цього ефекту може бути застосовано для підміни bootloader нальоту [13, 18]. Такий підхід дозволяє досягнути поставленої мети, а саме можливості перепрограмовувати окремі вузли у складі працюючої системи без їх вилучання.

**Виділення окремого вузла** для збереження резервних копій попередніх прошивок для використання програми-прошивки для програмування мікроконтролера у складі системи дозволяє заготовити колекцію для перепрограмування залежно від ситуації. Виділений мікроконтролер має містить діагностичні прошивки, які в ньому розташовані заздалегідь. За необхідності він також може , виконувати перезавантаження та перепрограмування окремих вузлів у складі системи, що необхідно для вирішення завдань діагностики або наступного системи.

Також у такому особливому вузли може зберігатися набір заготовлених програм для перебудови системи в залежності від поточного режиму. Перебудова передбачає можливість відновлення частини функцій чи передачі окремих функцій від одного набору мікроконтролерів іншому.

Розглянуті рішення та можливість комунікації між окремими вузлами дозволяють знизити вартість системи за рахунок використання чипів з нижнього цінового сегменту. Додатковою перевагою є можливість перепрограмування вузлів такої системи на льоту безпосередньо у складі системи та здатність виконувати самодіагностику проведеної реконфігурації. Додавання надмірності на рівні апаратного монтажу і можливості

перепрограмування прошивок дають можливість відновлення системи після певних видів відмов системи.

### 13.3.3. Аналіз переваг керування енергоспоживанням

Використання такої загальної єдиної лінії комунікації дозволяє виконувати пробудження окремих вузлів у складі системи і переводити в режим глибокого сну решту вузлів у випадку їх невикористання з метою економії енергії. Це дозволяє суттєво знизити енергоспоживання системи і загалом керувати енергоспоживанням при побудові систем, які містять елементи живлення.

Такий підхід також може бути актуальним для побудови розподілених систем моніторингу та інформування, в яких окремі вузли мають змішаний тип живлення з акумуляторним чи сонячним живленням. При такому способі функціонування системи існує необхідність періодичного пробудження її компонентів для комунікації чи діагностики потрібних вузлів.

Виробник мікроконтролерів надає можливість гнучко керувати енергоспоживанням складових мікроконтролера. Подібні операції можна виконувати шляхом вмикання чи вимикання складових мікроконтролера через регістри, які керують енергоживленням. До таких складових мікроконтролера слід віднести в першу чергу АЦП, таймери-лічильники, компаратори і живлення портів.

У випадку необхідності є можливість переводу всього мікроконтролера в режим глибокого сну для зниження енергоспоживання. Використання ліній комунікації з'єднаної з входами зовнішнього переривання дозволяє керувати пробудженням мікроконтролера у випадку появи сигналу на спільної лінії комунікації. Перевірка адреси окремого вузла дозволяє виконати його перехід у подальший сон, або активізувати у випадку збігу адреси. Реалізація такої підходу дозволяє виконувати пробудження з глибокого сну чи режиму зниженого енергоспоживання лише ті вузли, активність яких наразі необхідна.

### 13.3.4. Аналіз варіантів забезпечення безпеки та відмовостійкості рішень

**Шифрування обмінів між мікроконтролерами** може бути застосовано для підвищення ступеня захисту загального інтерфейсу між окремими вузлами системи [19]. Як захід для підвищення інформаційної стійкості системи бажано застосовувати один із криптопримітивів, що дозволить знизити вірогідність виникнення помилки під час обміну даними між окремими вузлами та виправити помилки у випадку використання оптичного чи радіоканалу. Використання криптопримітивів також може частково запобігти процесу реверс-інжинірингу, прослуховуванню та проникненню у склад системи зловмисника.

Найбільш перспективними для використання є неприведені поліноми. Вони застосовуються для побудови елементів з надмірністю і можливістю

побудови шифрування послідовності за допомогою генераторів з використанням LFSR. Можна також застосовувати криптопримітив на основі S-Box, що дає можливість змінювати послідовності бітів, що пересилаються, з використанням підстановки байтів за наявною таблицею.

У класичному випадку після застосування вихідної таблиці виконання реверс-інжинірингу даних є максимально складною задачею, що дозволяє передавати дані по відкритому каналу обміну. Це пояснюється тим, що набори даних, які розташовані послідовно, після проходження процедури підстановки являють собою білий шум.

Додатковим ступенем захисту може виступати модифікація закону формування повідомлень у складі обміну по заздалегідь встановленому для всіх вузлів алгоритму. Спільними рисами та сигналами можуть виступати паузи між обмінами даних та окремими словами, а в якості сигналів синхронізації – проміжки часу чи інтенсивність обмінів.

При такій організації системи Після відключення одного з елементів мережі на деякий період часу від загального обміну залишається можливість коректної роботи після його вмикання. Зберігається також можливість його синхронізації та автоналаштування способу дешифрування повідомлень у складі загальної лінії обміну без необхідності повторення усієї множини повідомлень.

Зазначений підхід з реалізацією захисту з використанням елементів криптографії дозволяє підвищити стійкість системи при тимчасовому відключення окремих вузлів системи і не дозволить використати криптоаналіз, спираючись лише на послідовність даних, які пересилаються.

Це вимагатиме від криптоаналітика використання додаткових способів аналізу, таких як аналіз за часом чи аналіз за енергоспоживанню. Здійснення таких видів аналізу потребує можливості безпосереднього доступу до системи і ускладнює процес реверс-інжинірингу системи і каналу обміну.

**Підвищення надійності системи** може бути досягнуто завдяки додаванню надмірності [20, 21]. Прикладом елементів надлишковості може бути використання множини ліній аналого-цифрового перетворювача у складі мікроконтролерів. Доступ до цих ліній виконується через елементи навантаження чи через лінії вимкненої периферії, які можуть бути використані для вирішення майже всіх завдань.

Для використання лінії мікроконтролера у такому режимі необхідно перевести лінію в режим зчитування рівня напруги і виконувати необхідні дії програмно.

Серед можливих прикладів використання слід відзначити аналіз послідовних інтерфейсів з низькою пропускною здатністю, аналіз фізичного монтажу і відсутності пошкоджень такого монтажу на виході. По таких лініях можлива також комунікація, хоча пропускна здатність зазвичай знижується порівняно із апаратною реалізацією інтерфейсу обміну.

Апаратні способи передавання даних можуть бути доповнені можливостями додаткового аналізу ліній і виводів, до яких апаратні блоки мікроконтролера підключені через навантаження чи периферійні пристрої.

Подібний спосіб резервування ускладнює реалізацію, оскільки потребує додаткових складних програмних рішень, хоча необхідність їх написання може виникнути вже після введення системи в експлуатацію.

Однак, знаючи про можливі складнощі з розробленням програмного забезпечення, можливо передбачити маловитратні апаратні заготовки для подальшої їх програмної активації. Для цього потрібний попередній аналіз і розуміння можливості написання такого коду до початку проектування апаратної схеми. Аналіз дає можливість вибрати пріоритетні варіанти використання портів мікроконтролерів для внесення надлишковості безпосередньо в сам монтаж і розведення схеми його підключення. Цей підхід дає можливість подальшого перепрограмування такого мікроконтролера із можливістю задіяти ці лінії в подальшому, і сприятиме полегшенню його перепрограмування. Можливість задіяти звільнені лінії в майбутньому дозволить відмовитися від використання додаткових елементів при виробництві пристрою.

Можливі кілька способів реалізації запропонованого рішення. Перший підхід полягає у надання доступу до окремих ліній через лінії підтяжки у складі мікроконтролера через ємнісні зв'язки і спільне навантаження. Другий підхід полягає в підвищенні струмових характеристик окремих виводів мікроконтролерів шляхом використання кількох ліній одного порту або ліній різних портів. Він є доречним оскільки за специфікацією виробника мікроконтролерів найбільші спільні показники струмів можливі для його виводів у випадку використання окремих портів. Цей підхід представляється більш перспективним, оскільки струми на виводах мікроконтролера матимуть найбільш близькі значення.

Додатковою можливістю можна розглядати використання таких виводів портів для комутації токів у навантаженні з регульованим споживанням, коли одна з ліній може бути задіяна с якоюсь підтяжкою через резистор з кінцевою лінією керування. У випадку відмови керуючого транзистора можливо живити цільовий пристрій невеликим струмом через лінію порту самого мікроконтролера. Такий спосіб можна застосувати там де існує можливість живлення струмом до 50 мА, наприклад, для живлення світлодіодів. Крім того, зберігається можливість використання пристрою за призначенням у режимі обмеженої функціональності але зі збереженням функцій. Подібна перебудова системи має перевести апаратне рішення у наступний стан неповної функціональності але зі збереженням набору функцій, що зберігає можливість експлуатації і використання компонента з частковими або обмеженими властивостями.

Перевагою такого підходу є можливість множинного і багаторівневого зменшення набору можливостей системи., що спричиняє деградація її

властивостей зі збереженням мінімальної функціональності або можливістю діагностування стану із подальшою перебудовою системи.

Для зниження трудовитрат при проектуванні таких систем може бути використано набір спеціальних пакетів чи елементів штучного інтелекту, що дозволить знизити трудовитрати розробника на аналіз для створення можливих елементів надмірності як під час трасування плат та і при побудові схем. Наступним кроком може виступати використання таких рішень для написання окремих версій проектів із виправленнями для відновлення функцій для кожного із можливих випадків відмови.

Такий підхід дозволяє повністю виключити або знизити вірогідність виникнення помилок чи дефектів під час проектування викликаних людським фактором і підготувати усю множину можливих програм-прошивок для кожного з сотні варіантів відмов. Доцільно зберігати ці прошивки у файловій системі у складі окремого флеш-накопичувача чи флеш-мікросхеми для можливості виявлення, діагностики та виправлення виявлених відмов.

Вартість такого рішення невелика, оскільки існує множина окремих мікросхем флеш-пам’яті з місткістю, яка суттєво перевищує місткість пам'яті окремого мікроконтролера. Такі мікросхеми пам'яті підтримують декілька інтерфейсів обміну, а їх вартість невелика порівняно із вартістю мікроконтролера.

Додавання такого елементу у склад схеми дає можливість доповнити схему здібністю перепрограмування в залежності від діагностованого виду відмови, що забезпечує перехід схеми без участі людини в один із необхідних режимів. З іншого боку, використання таких чипів дозволяє знизити вірогідність відмови за рахунок малої кількості виводів. До того ж використання технологій флеш-пам’яті сприяє стійкості до Single Event Upset відмов, які викликані нейтронним та космічним випромінюванням.

Розглянуті способи організації системи надають можливість відновлення системи після відмови чи збою по енергоживленню, оскільки флеш-пам’ять є енергонезалежною, що дозволяє при необхідності зберігати весь набір прошивок і програм незалежно від живлення.

## 13.4. Послідовність побудови систем з підтримкою дистанційної діагностики, перепрограмування і реконфігурації вузлів

Поєднання можливості організації взаємодії по одній лінії, можливості модифікації bootloader, можливості дистанційної діагностики і перепрограмування, можливості зміни функцій схеми завдяки внесеній надмірності дозволяє сформулювати процедуру розроблення відмовостійких систем з поступовою деградацією на базі готових індустріальних компонентів на базі мікроконтролерів, за рахунок можливості дистанційної діагностики, перепрограмування і реконфігурації окремих вузлів вбудованої системи.

Таким чином, для забезпечення можливості перепрограмування таких апаратних рішень на основі мікроконтролерів пропонується така послідовність:

1) модифікувати завантажувач для забезпечення нових додаткових функцій мікроконтролерів;

2) визначити вузли, при проектування яких потрібно реалізувати можливість їх перепрограмування, і модифікувати схему пристрою, яка таку можливість реалізує;

3) додати потенційно надлишкові елементів до схеми під час прототипування для забезпечення можливості подальшого ремонту;

4) використовувати перепрограмування окремих вузлів для проведення діагностики схеми;

5) використовувати однолінійний зв'язок в якості простого способу зв'язку між вузлами;

6) додати шифрування на основі LFSR або sBox для покращення захисту зв’язку через інтерфейс;

7) використовувати автоматизовані інструменти для розроблення апаратної та програмної частини із зазначеними функціями, що дозволить зменшити витрати на проектування.

## 13.5. Приклад застосування запропонованого методу

Описана концепція була реалізована для отримання масштабованих, відтворюваних і дешевих некритичних периферійних компонентів розумного будинку.

Для ефективного керування та організації роботи пристрою використовується бездротовий модуль із інтерфейсом UART. Пропускної здатності такого каналу достатньо для віддаленого керування та конфігурації пристрою. Це значно спростить завдання управління та координування, а також відкриє можливість організації спільної взаємодії групи пристроїв на низькому рівні. Це дозволяє оператору налагодити пристрій для роботи в різних умовах, розташувавши модулі у зручному для роботи порядку.

У наведеному прикладі розглядається панель керування з чотирма кнопками на основі фоточутливих діодів. Цей компонент системи побудований на чіпі Atmega8A зі спеціально розробленим завантажувачем, який дозволяє перепрограмувати цей мікроконтролер за допомогою загальної лінії зв'язку.

Периферійні компоненти, відповідальні за механічне перемикання контактів, мають обмежений життєвий цикл. Ця проблема основана на деградації металевих контактів перемикача.

Для поліпшення показників надійності реалізовані оптичні кнопки. Крім того, перед створенням модуля було змодельовано набір можливих відключень ліній у схемі. Модуль АЦП, внутрішньо підключений до порту C мікроконтролера, призначений для вимірювання сигналів від фотодіодів. Нормальний режим роботи представлений на рисунку 13.2. Для зменшення енергоспоживання резистори підтягування підключаються до виводу того ж порту. Цей вибір також є різновидом резервування.

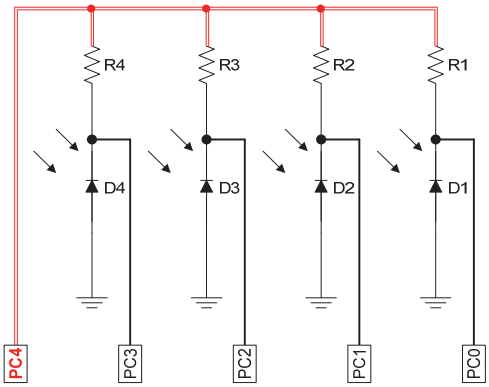



Рисунок 13.2 – Нормальна робота схеми панелі з 4 фотокнопками

Панель з фотодіодами і підтягуючими резисторами підключається до друкованої плати з мікроконтролером за допомогою дротової шини. Це найбільш вразливий до механічних пошкоджень елемент системи. Якщо один із виводів мікроконтролера відключений, цю проблему можна виявити за допомогою програми діагностики з перевіркою цілісності схеми.

Якщо відключеною лінією є загальна підтягуюча лінія, діоди можна отримати один за одним із підтягуванням через інші порти, підключені до діодів. Якщо відключеною лінією є лінія, пов'язана з діодом, сигнал від цього діода можна отримати через загальну підтягувальну лінію, налаштовану як вхід АЦП, як показано на рисунку 3.

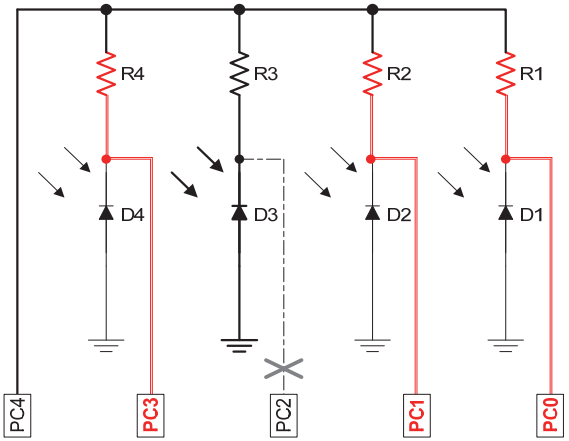



Рисунок 13.3 – Отримання сигналу з D3 після реконфігурації

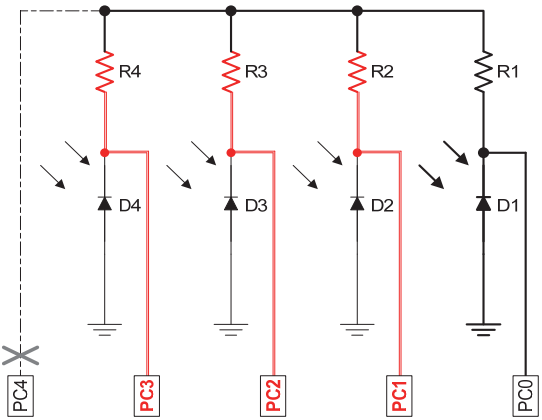



Рисунок 13.4 – Приклад конфігурації у разі відриву загальної лінії підтяжки

13.6. Висновки

Запропоновано підхід до розроблення відмовостійких систем з поступовою деградацією на базі готових індустріальних компонентів на базі мікроконтролерів, за рахунок можливості дистанційної діагностики, перепрограмування і реконфігурації окремих вузлів вбудованої системи.

Для цього проаналізовано можливі технічні рішення та теоретичні основи внутрішньосистемного перепрограмування вузлів системи на базі мікроконтролерів.

Представлено можливість використання лінії скидання reset для збереженням можливості перепрограмування мікроконтролерів за допомогою bootloader. Запропоновано ідею перепрограмування та діагностики вузлів системи по одній лінії за допомогою готових індустріальних компонентів на базі мікроконтролерів та засобів розробки, що дає можливість здешевити їх розроблення.

Запропоновані технічні рішення дають можливість підвищити надійність системи на основі вузлів з використанням мікроконтролерів, забезпечити керування енергоспоживанням цих вузлів та способи захисту лінії зв’язку на рівні bootloader.

Іншою перевагою підходу на основі дистанційної діагностики, перепрограмування і реконфігурації окремих вузлів можна вважати можливість організації зв’язку між мікроконтролерами за допомогою простішого інтерфейсу, який відповідає монтажному з’єднанню з використанням однієї лінії зв'язку. Це найпростіший у реалізації спосіб з'єднання вузлів на відміну від складних протоколів взаємозв’язку мікроконтролерів з використанням інтерфейсів I2C, CAN, багатокористувацький SPI чи модифікації RS485.

Для систем не критичного застосування, є доцільним використання рішень на основі ліній комунікації з описаними властивостями. Запропонований підхід може бути корисним для вирішення завдань моніторингу, зважаючи на простоту реалізації обміну і відносну нескладність апаратної реалізації.

Запропоновано семикрокову процедуру розроблення таких систем. Наведено практичний приклад реалізації концепцій розглянутих у статті, включаючи її реконфігурацію; наводиться схема пристрою до і після виконання його реконфігурації.

Запропоновані елементи методу дистанційної діагностики, перепрограмування і реконфігурації окремих вузлів вбудованої системи на базі мікроконтролерів.

Практичне значення даного дослідження полягає у можливості побудови і налаштування системи з великої кількості окремих вузлів на базі мікроконтролерів. Цей крок дозволяє розглядати побудову систем із функціями цифрових пристроїв з використанням адитивних технологій та 3d друку.

Література

1. Plakhteyev, A. Edge computing for IoT: An educational case study [Text] / A. Plakhteyev, A. Perepelitsyn, V. Frolov // Proceedings of 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies, DESSERT 2018. – 2018. – P. 130–133. DOI: 10.1109/DESSERT.2018.8409113.

2. Diversity metric evaluation considering extended NUREG-7007 diversity classification [Text] / V. Duzhyi, V. Kharchenko, A. Panarin, D. Rusin // Proceedings of 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies, DESSERT 2018. – 2018. – P. 21–25. DOI: 10.1109/DESSERT.2018.8409092.

3. Vdovichenko, O. Technologies for building systems of remote lining of communication lines: a practical example of implementation [Text] / O. Vdovichenko, A. Perepelitsyn // Radioelectronic and Computer Systems. – 2021. – No. 2. – P. 31–38. DOI: 10.32620/reks.2021.2.03.

4. Поночовний, Ю. Л. Методологія забезпечення гарантоздатності інформаційно-керуючих систем з використанням багатоцільових стратегій обслуговування [Текст] / Ю. Л. Поночовний, В. С. Харченко // Радіоелектронні і комп'ютерні системи. – 2020. – № 3. – С. 43–58. DOI: 10.32620/reks.2020.3.05.

5. Исследование подходов к построению орбитальной вычислительной сети спутниковой системы Интернета Вещей [Текст] / М. Е. Ильченко, Т. Н. Нарытник, В. И. Присяжный, С. В. Капштык, С. А. Матвиенко // Авіаційно-космічна техніка і технологія. – 2019. – № 8. – С. 138–151. DOI: 10.32620/aktt.2019.8.21.

6. Вдовіченко О. О. Розробка роботизованої системи для дистанційного прокладання ліній комунікації [Текст] / Вдовіченко О. О., Перепелицин А. Є. // Проблеми інформатизації : восьма міжнародна науково-

технічна конференція. – 2020. – Том 1. – С. 27. Available at: https://repository.kpi.kharkov.ua/server/api/core/bitstreams/f3dfa2fd-523a-4079-b501-ca219c50f615/content (accessed - 18. 04. 2023).

7. Robotics News: Inchworm Available at: https://robotics.ua/news/prototypes/1752-incredibly_fast_worms_inchworm_robots_work_only_ on_one_engine (accessed - 18. 04. 2020).

8. Robotics News: Bionic Wheel Bot Available at: https://robotics.ua/news/prototypes/7080-spider_robot _festo_bionicwheelbot_video (accessed - 18. 04. 2020).

9. IQBX - електромеханічний конструктор [IQBX - Electromechanical Designer] Available at: https://boomstarter.ru/ projects/ 945280/161131 (ac-cessed - 12. 09. 2020).

10. Конструктор LEGO [Constructor LEGO] Available at: https://www.lego.com/en-us/themes/technic (accessed -12. 09. 2020).

11. Arduino Platform Available at: https://learn.sparkfun.com/tutorials/what-is-an-arduino/all (accessed - 12. 09. 2020).

12. Блог технической поддержки [Technical support blog] Available at: http://mypractic.ru/urok-48-obmen-dannymi-mezhdu-platoj-arduino-i-kompyuterom - cherez-interfejs-uart.html (accessed - 04. 03. 2021).

13. Lewandowski, M. Dedicated AVR Bootloader for Performance Improvement of Prototyping Process [Text] / M. Lewandowski, T. Orczyk, P. Porwik // 2017 MIXDES – 24th International Conference "Mixed Design of Integrated Circuits and Systems". – 2017. – P. 553–557. DOI: 10.23919/MIXDES.2017.8005274.

14. Embedded Hardware Testing Using Bootloader [Text] / A. Rath, D. Roy, D. Teja, G. Kumar // International Conference on Smart Electronics and Communication, ICOSEC 2020. – 2020. – P. 1–6. DOI: 10.1109/ICOSEC49089.2020.9215327.

15. ATmega8A Data Sheet [Online]. – Microchip Technology Inc., 2020. – 324 p. – Available at: https://ww1.microchip.com/downloads/en/DeviceDoc/ATmega8A-Data-Sheet-DS40001974B.pdf. – 19.10.2022.

16. Kolisnyk, M. Vulnerability analysis and method of selection of communication protocols for information transfer in Internet of Things systems [Text] / M. Kolisnyk // Radioelectronic and Computer Systems. – 2021. – No. 1 – P. 133–149. DOI: 10.32620/reks.2021.1.12.

17. IEEE P802.3cg 10Mb/s Single Pair Ethernet: A guide [Text] / G. Zimmerman, P. Jones, J. Lewis, P. Beruto, S. Graber, H. Stewart. – Cisco Systems Inc., 2019. – 40 p.

18. Sebastian, A. Design of a Dynamic Boot Loader for Loading an Operating System [Text] / A. Sebastian, S. Sankar // Journal of Computer Science. – 2019. – Vol. 15, no. 1. – P. 190-196. DOI: 10.3844/jcssp.2019.190.196.

19. Technique for IoT malware detection based on control flow graph analysis [Text] / K. Bobrovnikova, S. Lysenko, B. Savenko, P. Gaj, O. Savenko // Radioelectronic and Computer Systems. – 2022. – No. 1 – P. 141–153. DOI: 10.32620/reks.2022.1.11.

20. FPGA platform-based NPP I&C systems: Case study of diversity assessment and selection [Text] / V. Kharchenko, E. Brezhnev, V. Sklyar, V. Duzhyi // Proceedings of the 9th International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies, NPIC and HMIT 2015. – 2015 — P. 93-102.

21. Tyurin, S. Hyper redundancy for super reliable FPGAs [Text] / S. Tyurin // Radioelectronic and Computer Systems. – 2021. – No. 1. – P. 119-132. DOI: 10.32620/reks.2021.1.11.

# 14. ЗНАННЯ-ОРІЄНТОВАНІ МЕТОДИ ТА ЗАСОБИ АВТОМАТИЗАЦІЇ СИНТЕЗУ ТЕСТІВ

**К. С. Гайдук, І. М. Клюшніков, О. І. Морозова**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 14.1. Вступ

Щоденно у світі з'являється багато нової інформації, все частіше виникають нові предметні області, які характеризується насамперед спеціальними знаннями, що описують об'єкти цієї галузі та їх властивості. Практичне використання таких знань є долею експертів. Власне, у володінні такими знаннями і є професійна компетентність експерта. Однак залишатися всезнаючим експертом у наші дні стає все складніше, оскільки прискорюється зміна технологій та змісту експертних знань. У тому числі це відбувається тому, що практично у будь-якій предметній галузі, у будь-якій галузі генерується величезна кількість даних.

Помічено, що кількість даних збільшується за експонентою, тобто чим більше стає даних, тим швидше швидкість збільшення їхньої кількості. Це дуже серйозний тренд – і психологічно, і технологічно. Такий великий обсяг нової інформації стає дедалі складніше передавати, обробляти, зберігати, попри суттєве збільшення продуктивності устаткування. Але найбільша проблема полягає не в розмірі даних, а в їх неструктурованості . Дані з'являються з різних джерел, у різних форматах, у різний час. Тому перед використанням у практичних завданнях дані необхідно впорядковати, перетворити (привести) у форму, ефективну для зберігання та використання.

Традиційним способом зберігання та використання даних є реляційні бази даних, у яких дані зберігаються у вигляді пов'язаних таблиць. Однак не завжди табличні дані ефективні у використанні, тому почали з'являтися альтернативні форми, наприклад, системи організації знань (Knowledge Organization Systems , KOS), як правило, що базуються на графах знань.

Для зберігання знань можуть використовуватися різні структури:

керовані словники: забезпечують спосіб організації знань для подальшого пошуку, використовуються в схемах предметної індексації, предметних рубриках, тезаурусах, таксономії та інших системах організації знань,

тезауруси: поєднують терміни в групи за певною ознакою, наприклад, з урахуванням схожості (синоніми),

таксономії: категоровані слова, упорядковані за ієрархічною ознакою,

онтології: формальний опис знань з якогось домену (предметної області) з урахуванням наявних складних правил та зв'язків між елементами, що дозволяє зробити автоматичне вилучення знань,

датасети: набори машинних даних.

Онтоло́гія, це формалізоване представлення знань про певну предметну область (середовище, світ), придатне для автоматизованої обробки. Онтологію неодмінно супроводжує деяка концепція цієї області. Найчастіше ця концепція виражається за допомогою визначення базових об'єктів (індивідуумів, атрибутів, процесів) і відношень між ними. Визначення цих об'єктів і відношень між ними зазвичай називають концептуалізацією, сутність якої полягає у визначені базису для моделювання цієї області знань, визначені протоколів для взаємодії між агентами, які використовують знання з цієї області, і, нарешті, встановленні домовленостей про представлення теоретичних основ даної області знань.

Онтології служать для організації знань і застосовують у тих галузях, де потрібно виявити нові факти, виявити приховані взаємозв'язки між елементами (наприклад, рекомендаційні та експертні системи). Це альтернатива класичним базам даних, які використовують «гіпотезу закритого світу», коли все, чого немає у базі даних – немає. На противагу цьому онтології використовують «гіпотезу відкритого світу», тобто якщо чогось немає в основі знань, то це не обов'язково не існує, а просто не описано досі.

Серед ріжних моделей подання знань [1], таких яких логічна, продукційна, семантичні мережі та ін., онтологічна модель характеризується відносною легкістю побудови, дослідженим механізмом логічного виводу, високим рівнем структурованості, легкістю подання родо-видових відносин, розвиненими виразними можливостями. Системи організації знань з урахуванням онтологій дуже поширені й у багатьох галузях. Найяскравіший приклад – це knowledge graph для пошуку інформації в Інтернеті, завдяки цій технології якість пошуку стала дуже високою.

Ситуативне формування структури складних систем під конкретні завдання може бути забезпечене за рахунок отримання великої кількості різноманітних рішень у вибраній предметній галузі. І перспективним напрямом формалізації таких знань є використання онтологій.

У таких випадках онтологія визначає спільну мову для опису предметної галузі задачі синтезу та включає машинно-інтерпретовані формулювання основних понять та відносин між ними.

Обсяг онтології визначається ступенем деталізації. Верхні онтології, також відомі як онтології верхнього рівня, онтології верхнього рівня або фундаментальні онтології, є загальними онтологіями, які застосовуються до різних предметних областей. Онтології предметної області описують словник конкретної предметної області з певною точкою зору та фактичними даними. Основні еталонні онтології є стандартизованими або стандартні де-факто онтології, що використовуються різними групами користувачів для інтеграції їх різних точок зору на предметну область шляхом злиття декількох онтологій предметної області (рис. 14.1).

В загальному випадку, онтологія - це кортеж вигляду <C, R, T>, в якому C - множина понять предметної галузі, R - множина відношень між поняттями, T - множина правил.

Онтології зазвичай містять класи (поняття), екземпляри цих класів, їхні атрибути (властивості) та значення цих властивостей, а також відношення між класами та екземплярами класів. Крім того, онтологія може містити певні обмеження на використання класів та їх відношень.

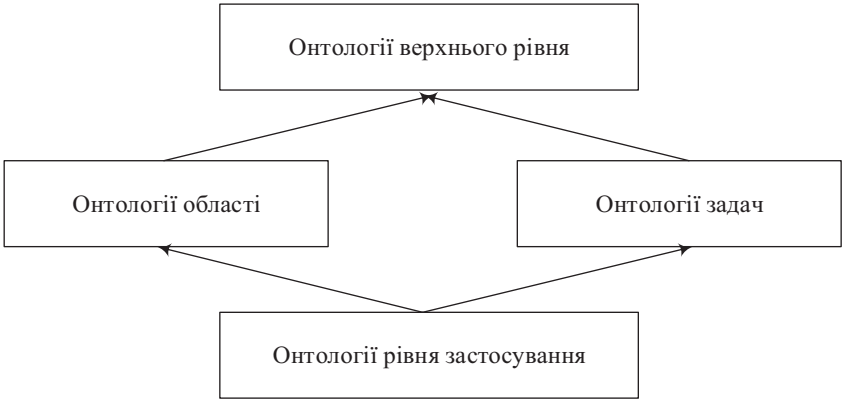



Рисунок 14.1 – Ієрархія типів онтологій

Екземпляри (англ. instances) або індивіди (англ. individuals) — це основні, низькорівневі компоненти онтології. Екземпляри можуть являти собою як фізичні об'єкти (люди, будинки, планети), так і абстрактні (числа, слова). Щиро кажучи, онтологія може обійтися й без конкретних об'єктів. Однак, однією з головних цілей онтології є класифікація таких об'єктів, тому вони також включаються.

Поняття (англ. concepts) (або класи (англ. classes)) — абстрактні групи, колекції або набори об'єктів. Вони можуть містити в собі екземпляри, інші класи, або ж сполучення й того, і іншого. Приклад:

Поняття «люди», вкладене поняття «людина». Чим є «людина» — вкладеним поняттям, чи екземпляром (індивідом) — залежить від онтології.

Поняття «індивіди», екземпляр «індивід».

Об'єкти в онтології можуть мати атрибути. Кожен атрибут має принаймні ім'я й значення, і використовується для зберігання інформації, що специфічна для об'єкта й прив'язана до нього. Наприклад, об'єкт «Кафедра» може має такі атрибути:

*Назва*: Кафедра компьтерних систем мереж і кідербезпеки.

*Кількість викладачів*: 40.

*Кількість аудиторій*: 20.

*Спеціальності підготовки*: 123, 125.

Значення атрибута може бути складеним типом даних. У цьому прикладі значення атрибута, що називається *Спеціальності підготовки*, є списком значень простих типів даних.

Важлива роль атрибутів полягає в тому, щоб визначати залежності (відношення) між об'єктами онтології. Зазвичай відношенням є атрибут, значенням якого є інший об'єкт.

В математичному підставі онтології лежить так звана дескриптивна логіка (розділ математики), яка передбачає, що будь-яка інформація, висловлена природною мовою, може бути представлена у вигляді ланцюжка триплетів (рис. 14.2)


Рисунок 14.2 – Приклад триплету

Онтологія представляється як граф, вершини якого це сутності, а ребра – відносини між сутностями. Вважається, що будь-яке твердження природною мовою можна подати у вигляді простих пропозицій, з яких можна отримати сутності та відносини між ними.

Властивості в онтологіях представляють відносини. Існує два основних типи властивостей: властивості об'єктів (Object properties) та властивості типів даних (Data properties). Властивістю об'єкта є відносини між двома індивідами.

Властивості об'єкті пов'язують індивідів. В онтологіях, що описуються мовою OWL, є третій тип властивості – властивості анотації. Властивості анотації можуть використовуватися для додавання інформації (метадані дані про дані) для класів, окремих індивідів та властивостей об'єктів/типів даних. На рис. 14.3 представлені приклади властивостей різних типів.

Зворотні властивості (Inverse)

Кожна властивість об'єкта може мати відповідну зворотну властивість. Якщо деякі властивості пов'язує індивіда ***a*** з деяким індивідом **b** , його зворотне властивість пов'язує індивід **b** з індивідом ***a*** . Наприклад, на рис.14.4 показано властивості *hasPayload* та його зворотну властивість *deployedOn* — якщо UAV *hasPayload*, то зі зворотної властивості можна зробити висновок, що Camera *deployedOn* UAV.

Властивості можуть мати різні характеристики, які описані нижче.

Функціональні властивості (Functional).

Якщо властивість є функціональною, то для даного індивіда може існувати не більше одного індивіда, який має відношення до першого індивіда через цю властивість. Функціональні властивості також відомі як однозначні властивості, а також як індивідуальні особливості індивіда.

Зворотні функціональні властивості (Inverse Functional).

Якщо властивість має зворотню функціональність, це означає, що властивість є зворотньо функціональною. Для конкретного індивіда може бути безліч індивідів, що належать до першого індивіда через цю властивість.

Транзитивні властивості (Transitive).

Якщо транзитивна властивість і властивість зв'язує індивіда ***a*** і індивіда ***b*** , а також індивіда ***b*** пов'язує з індивідом ***c*** , то ми можемо вивести, що цей індивід ***a*** пов'язаний з індивідом **c** через цю властивість. На рис. 14.5 показано приклад транзитивної властивості *belongTo* . Якщо індивід *PC-1* належить до класу *RotaryWingUAV*, а *RotaryWingUAV* належить до класу *UAV* , то можна зробити висновок що *PC-1* належить до класу *UAV* – це позначено пунктирною лінією.

Симетричні властивості ( Simmetric )

Якщо властивість ***p*** симетрична, і властивість пов'язує індивіда ***a*** з індивідом ***b*** , то індивід ***b*** пов'язаний також з індивідом ***через*** властивість ***p*** . Рис. 6 показує приклад симетричної властивості. Якщо індивід Mavic пов'язаний з індивідом PC-1 через властивість *hasSameCharacterictics*, то можна вважати, що PC-1 повинна бути також пов'язана з Mavic через властивість *hasSameCharacterictics*. Таким чином, симетрична властивість є зворотньою (рис. 14.6).

Асиметричні властивості (Asimmetric)

Якщо властивість ***p*** асиметрична, і властивість пов'язує індивіда ***i*** з індивідом ***b*** , то індивід ***b*** може бути пов'язані з індивідом ***a*** через властивість ***p***.

Приклад асиметричної властивості. Якщо індивід Валькірія пов'язаний з індивідом Літаюче Крило через властивість *належитьДо* , то можна дійти висновку що Літаюче Крило не пов'язане з Валькірія через властивість *належитьДо* .

Рефлексивні властивості (Reflexive)

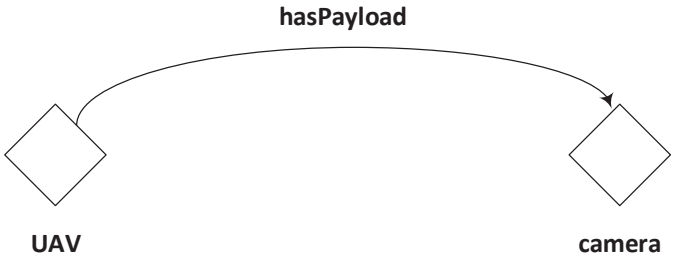
Властивість ***P*** називається рефлексивною, коли індивід ***a*** має бути пов'язаний із собою. Приклад. Індивід PC-1 повинен мати відношення до себе за допомогою властивості знати. Іншими словами PC-1 має знати себе. Однак, крім того, можливо для PC-1 знати про інші БПЛА .

Ірефлексивні властивості (Irreflexive)

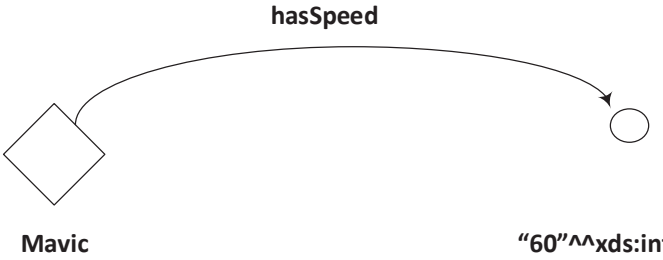
Якщо властивість ***p*** ірефлексивна , то вона може бути охарактеризована як властивість, яка пов'язує індивіда ***a*** з індивідом ***b***, де індивід ***a*** та індивід ***b*** обов'язково різні.

Домени та діапазони властивостей

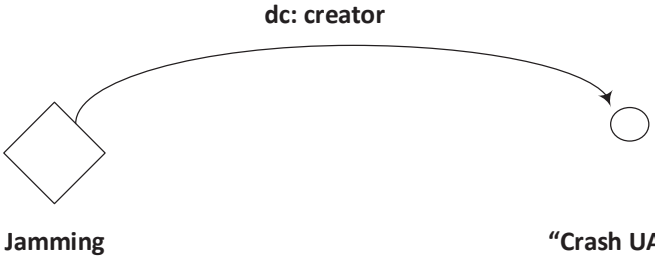
Властивості можуть мати домени (Domains) та діапазони (Ranges). Властивості пов'язують класи (індивідів) із доменів із класами (індивідами) в діапазонах. Наприклад. у онтології Система моніторингу , властивість *hasPayload* ймовірно зв'яже індивідів, що належать до класу UAV с індивідами, що належать до Навантаження. У цьому випадку UAV є доменом властивості *hasPayload*, а діапазоном – Payload, як це показано на рис. 14.7.




Властивість об'єкту, що
пов'язує індивідів




Властивість типу даних, що пов'язує
індивіда з числовим значенням 60 в
форматі xds:integer




Властивість анотація, що пов'язує клас
**Jamming** з символьним рядком **“Crash UAV”**

Рисунок 14.3 – Типи властивостей

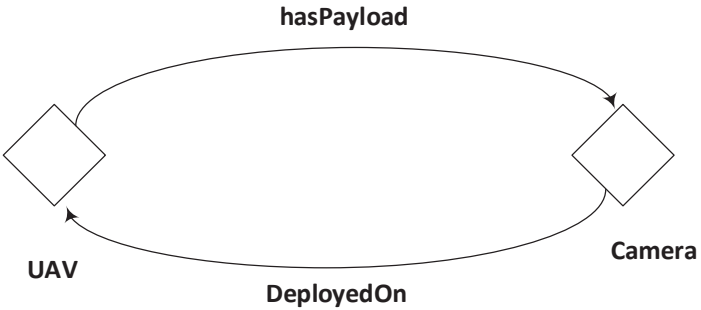



Рисунок 14.4 – Приклад зворотної властивості

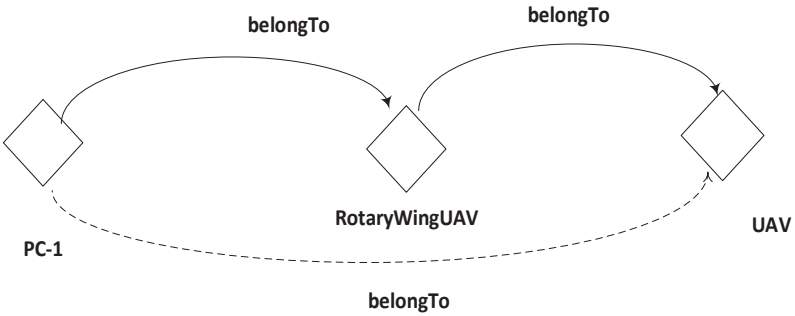



Рисунок 14.5 – Приклад транзитивної властивості

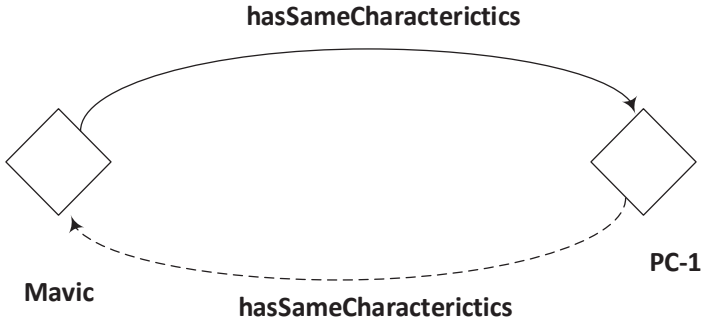



Рисунок 14.6 – Приклад симетричної властивості

Властивості типів даних (Data Properties) є властивостями, значеннями яких є літерали. Для цих властивостй також визначаються Domains і Ranges. Domain - це набір класів, до екземплярів яких застосовується дана властивість. Range – це тип значень, які властивість може набувати. Відповідно значення для Domain вибирається зі списку класів, що існує в нашій моделі, а Range – з певних стандартом типів даних (використовуються типи даних XSD). І Domain'ів, і Range у властивості може бути кілька.

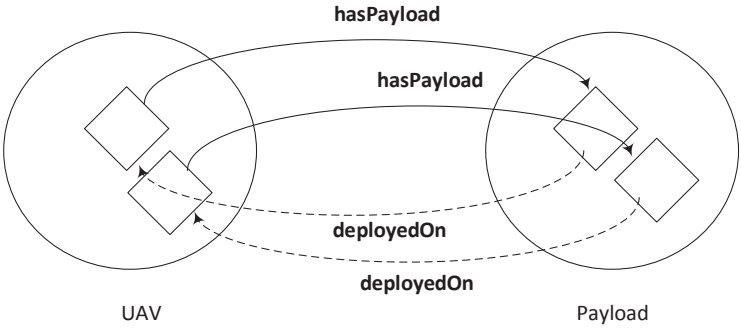



Рисунок 14.7 – Домен та діапазон для властивості *hasPayload* та його зворотньої властивості *deployedOn*

Для властивостей обох типів можна також встановити обмеження на кількість значень, які може мати дану властивість для конкретного об'єкта (наприклад, імен у людини може бути кілька, а дата народження – лише одна). І тому використовується поняття Cardinality. В такому випадку потрібно вибрати властивість, на яку накладається обмеження, потім клас(и) або тип(и) значень, що приймаються. За замовчуванням будь-яка властивість може мати будь-яку кількість значень кожного об'єкта.

Для опису онтологічних моделей, за останні три десятиріччя було створено низку формальних мов [2], окремо з яких варто відзначити наступні: RDF (Resource Description Framework) - дозволяє подавати знання предметної галузі у вигляді триплетів, однак не має стандартних засобів мови для опису таксономій (дерев родо-видових відносин); RDFS (RDF Schema) - має засоби для опису таксономій та анотування сутностей, проте не дозволяє опосередковано описувати класи (множини сутностей); OWL (Ontology Web Language) - позбавлена недоліків попередніх двох.

Файл онтології OWL можна визначити за допомогою URI онтології, предикату rdf:type і поняття Ontology зі словника OWL (лістинг 14.1) [2x].

Лістинг 14.1. Файл OWL, визначений як онтологія

*@ prefix rdf: < http://www.w3.Org/1999/02/22-rdf-syntax-ns# > .*
*@ prefix owl: < http://www.w3.Org/2002/07/owl# > .*

*< http://example.com/cyberontology.owl > owl:Ontology.*

Класи можуть бути оголошені з використанням rdf:type та концепції Class із словника OWL (лістинг 14.2).

Лістинг 14.2. Оголошення класу OWL

*:UAV a owl:Class.*

Відносини між суперкласом та підкласом можна визначити за допомогою властивості subclassOf зі словника RDFS (лістинг 14.3).

Лістинг 14.3. Відношення між класами

*:RotaryWingUAV rdfs:subClassOf :UAV .*

В OWL властивості, що визначають відносини між сутностями, оголошуються за допомогою owl: ObjectProperty, а ті, які надають властивості значенням властивостей або діапазонам значень – за допомогою owl:DatatypeProperty , як показано на прикладах у лістингу 14.4.

Лістинг 14.4. Властивість об'єкта та оголошення властивості типу даних

*:connectedTo a owl:ObjectProperty .*
*:hasIPAddress a owl:DatatypeProperty.*

Імена понять (класів, індивідуумів) в OWL зазвичай записуються в PascalCase, в якому слова створюються шляхом поєднання слів з великими літерами, а імена властивостей OWL зазвичай записуються в camelCase, тобто складові слова або фрази записуються таким чином, що кожне слово або абревіатура починаються з великою літерою (середня велика).

Окремі індивіди можуть бути оголошені з використанням терміну, що називаєтьсяіндивідуальним з словника OWL (лістинг 14.5).

Лістинг 14.5. Оголошення індивіда

*:PC-1 a owl:namedIndividual.*

Індивіди також можуть бути оголошені як екземпляр класу, як показано в першому прикладі. В OWL також доступні багато інших конструкторів, таких як атомарне і складне заперечення понять, перетин понять, універсальні обмеження, обмежена екзистенційна квантифікація, транзитивність, ієрархії ролей, зворотні ролі, функціональні властивості, типи даних, номінали та обмеження кардинальності. У другій версії OWL, OWL 2 вони доповнюються об'єднанням кінцевого набору складних включень ролей і рольової ієрархії.

Спільно з OWL для опису правил онтології використовується мова SWRL (Semantic Web Rule Language).

Мова SWRL дозволяє описувати відносно прості правила, подібні до тригерів у базах даних, призначені для деякої автоматизації додавання в онтологію властивостей екземплярів та відношень [4]. Дана мова не дозволяє задавати правилам імена, "викликати" одні правила із тіла інших (подібно до функцій), створювати власні предикати, реалізовувати низку відносно простих алгоритмів. Обмеженість мови SWRL добре ілюструється поданим нижче прикладом.

Нехай, дано невелику онтологію, ієрархію класів якої наведено на рис. 14.8. Бачимо клас "хімічні речовини" (ChemicalSubstances) з дочірніми класами "прості речовини" (SimpleSubstances), "хімічні сполуки" (ChemicalCompounds) та "сплави" (Alloys). Також видно клас "хімічні елементи" (ChemicalElements) із дочірнім класом "атом елемента в хімічній речовині" (AtomElementInChemicalSubstance).

Онтологія містить наступні екземпляри: H (водень) та S (сірка), як екземпляри класу "хімічні елементи"; H_In_H2S (водень у складі сірководню) та S_In_H2S (сірка у складі сірководню) як екземпляри класу "атом елемента в хімічній речовині", пов'язані із H та S відповідно об'єктним відношенням concretelyExpresses ("конкретна реалізація"); H2S (сірководень) як представник класу "хімічні сполуки", пов'язаного відношеннями hasPart із H_In_H2S та S_In_H2S.

Екземпляри класу "хімічні елементи" мають атрибут atomicMass (атомна маса), екземпляри класу "атом елемента в хімічній речовині" - атрибут numberOfAtoms (кількість атомів), а екземпляри класу "хімічні речовини" - атрибут molecularMass (молекулярна маса).

Для автоматизації розрахунку молекулярної маси хімічних речовин в онтології можна записати наступне SWRL-правило:

ChemicalCompounds(?X), hasPart(?X, ?A1), hasPart(?X, ?A2), numberOfAtoms(?A1, ?C1), numberOfAtoms(?A2, ?C2), concretelyExpresses(?A1, ?E1), concretelyExpresses(?A2, ?E2), atomicMass(?E1, ?M1), atomicMass(?E2, ?M2), notEqual(?M1, ?M2), multiply(?P1, ?M1, ?C1), multiply(?P2, ?M2, ?C2), add(?M, ?P1, ?P2) -> molecularMass(?X, ?M).

Результатом застосування вищенаведеного правила до онтології буде автоматичний розрахунок значення атрибута molecularMass для всіх хімічних сполук, які складаються з двох елементів. Зокрема, molecularMass(H2S) = 1 * 2 + 32 * 1 = 34 а. о. м.

Для автоматизації розрахунку молекулярної маси сполук, що складаються з трьох, чотирьох і т. д. елементів, потрібно складати відповідні SWRL-правила. Очевидно, що це дуже трудомісткий та не універсальний підхід. Крім того, наявність таких правил в одній онтології буде призводити до конфліктів при автоматичному логічному виводі. Одним з можливих рішень зазначеної проблеми може бути використання запитів до онтології SPARQL 1.1 Update [5], з допомогою яких можна виконати потрібні обчислення, та додати результати в онтологію. Так, в [6] наведено приклад SPARQL-запиту для розрахунку

молекулярної маси речовин, що складаються з довільної кількості елементів, та додавання результатів обчислень в онтологію. Проте, навіть використання SPARQL-запитів не є універсальним рішенням, з огляду на існування сполук з формулами виду $Ca(OH)_2$, $CuSO_4*5H_2O$ та ін. Очевидно, що можливостей правил SWRL та SPARQL-запитів недостатньо для адекватного (за об'ємом коду та його складністю) вирішення навіть такої тривіальної задачі як розрахунок молекулярної маси хімічної сполуки.

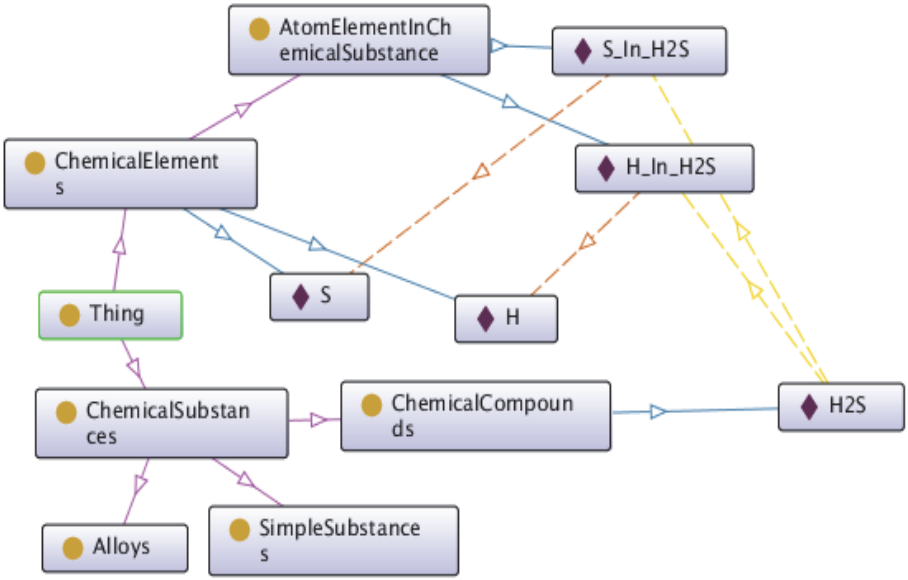



Рисунок 14.8 – Граф онтології

Певним рішенням вищезазначеної проблеми є опис правил онтології мовою Prolog [7], проте, в більшості випадків, написання таких правил є досить трудомістким, і отримуваний код (базу правил у вигляді скрипта) важко супроводжувати. Таким чином, актуальною є розробка мови опису правил онтології, яка поєднувала би простоту синтаксису SWRL з обчислювальною повнотою Prolog.

Метою досліджень є розроблення мови опису правил онтологій, що буде поєднувати простоту синтаксису SWRL та обчислювальні можливості Prolog. Для досягнення поставленої мети необхідно виконати наступні завдання:

1. Виконати аналіз синтаксичних, виразних та обчислювальних можливостей мов RDF, RDFS, OWL, SPARQL, Prolog.
2. Описати формальну граматику мови опису правил онтології. Використовуючи бібліотеку Parglare мови програмування Python, реалізувати

лексико-синтаксичний аналізатор для отриманої мови, з подальшою трансляцією в правила на мові Prolog.

3. Описати формальну граматику підмножини мови Prolog, в яку виконується відображення правил на розробленій раніше мові. Реалізувати лексико-синтаксичний аналізатор для вказаної підмножини мови Prolog, з подальшою трансляцією в правила на розробленій раніше мові.

Розробити редактор правил з веб-інтерфейсом та клієнт-серверною архітектурою.

## 14.2. Розроблення мови опису правил онтології Thoth

Запропонована мова базується на синтаксисі SWRL, та описується формальною граматикою, поданою в [6]. Дана мова підтримує більшість предикатів із SWRL, і може бути розширена правилами-предикатами довільної складності. Наприклад, в поточній версії мови наявні предикати sppco(X, Rel, P1, P2, S) - розрахунок суми S добутків значень атрибутів P1 і P2 для об'єктів, з якими об'єкт X пов'язаний відношенням Rel, та spco(X, Rel, P, S) - розрахунок суми S значень атрибутів P для об'єктів, з якими об'єкт X пов'язаний відношенням Rel.

Перш, ніж розглянути приклади практичного застосування правил на мові Thoth, введемо позначення: $E^1_{k1}...E^n_{kn}$ - загальна формула сполук на кшталт $Al_2O_3$, де $E^i$ - символ елементу, ki - кількість атомів відповідного елементу в молекулі сполуки; $E^1_{k1}...E^n_{kn}*qH_2O$ - загальна формула кристалогідратів з q молекулами води; $R^1_{k1}...R^n_{kn}$ - загальна формула сполук, що складаються з радикалів (наприклад, $CH_3COOH$, що складається з радикалів $CH_3$- та -COOH). Нижче наведено правила розрахунку молекулярної маси для сполук із різними загальними формулами (префікси виду ns: та dul: відповідають просторам імен в онтології):

- ns:ChemicalCompounds(X), sppco(X, dul:hasPart, ns:atomicMass, ns:numberOfAtoms, M) -> ns:molecularMass(X, M) - для сполук із загальною формулою $E^1_{k1}...E^n_{kn}$;
- ns:CrystallineSolids(X), dul:hasPart(X, S), ns:Salts(S), ns:molecularMass(S, M), dul:hasPart(X, H), ns:molecularMass(H, HM), ns:numberOfMolecules(H, K), multiply(HMK, HM, K), add(XM, M, HMK) -> ns:molecularMass(X, XM) - для кристалогідратів ($E^1_{k1}...E^n_{kn}*qH_2O$);
- ns:Radicals(X), sppco(X, dul:hasPart, ns:atomicMass, ns:numberOfAtoms, M) -> ns:molecularMass(X, M) - для радикалів;
- ns:ChemicalCompounds(S), !ns:CrystallineSolids(S), spco(S, dul:hasPart, ns:molecularMass, M) -> ns:molecularMass(S, M) - для сполук із загальною формулою $R^1_{k1}...R^n_{kn}$.

Правила, записані мовою Thoth, підлягають лексико-синтаксичному аналізу та трансляції в правила на мові Prolog (рис. 14.9). Текст утворюваних за результатами трансляції правил відповідає підмножині мови Prolog, яка описується формальною граматикою, поданою в [6]. Вказана граматика

дозволяє виконувати лексико-синтаксичний аналіз сгенерованого скрипта на Prolog, та зворотну трансляцію в правила на мові Thoth. Зазначені етапи лінгвістичної обробки правил реалізовано в розробленому редакторі правил з веб-інтерфейсом [6] (рис. 14.10). Застосування правил на Prolog до онтології може бути реалізоване з використанням бібліотеки semweb/rdf11 [7].

Наукова новизна отриманих результатів полягає в тому, що запропонований підхід до описання правил онтології дозволяє поєднати простоту синтаксису мови SWRL із обчислювальними можливостями мови Prolog.

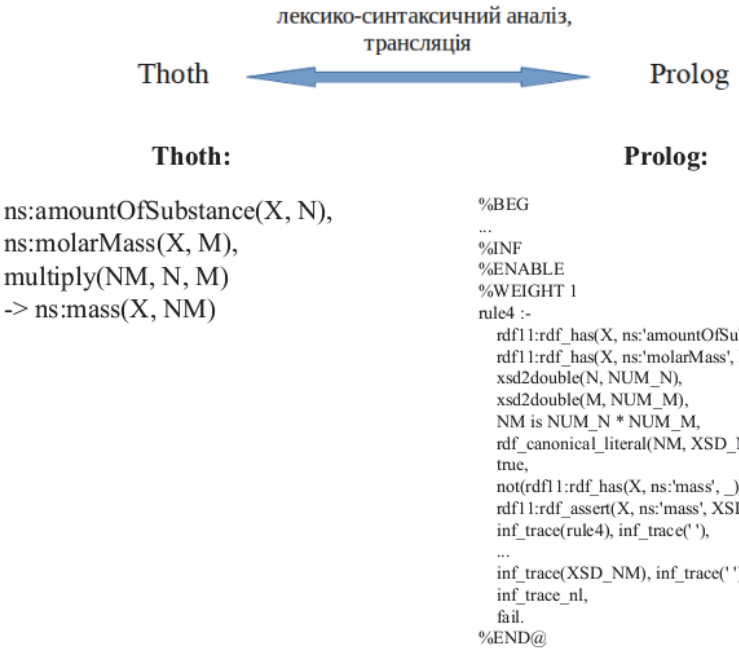



Рисунок 14.9 – Переклад правил онтології

Практичне значення отриманих результатів полягає в тому, що розроблено мову опису правил онтології, яка поєднує простоту синтаксису SWRL та обчислювальні можливості Prolog, а також розроблено відповідний редактор правил у вигляді веб-додатка.

Результати наукових досліджень доповідались та обговорювались на XV Міжнародній науково-практичній конференції "Інформаційні технології і автоматизація – 2022" (м. Одеса, 20-21 жовтня 2022 р.) [8].

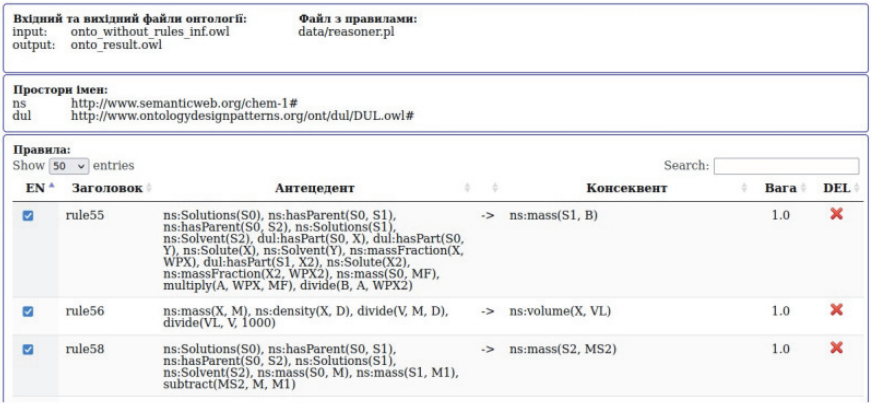



Рисунок 14.10 – Веб-інтерфейс редактора правил

## 14.3 Висновки

У розділі представлено основні поняття, що стосуються розробки онтологічних моделей. Розробка онтологій може забезпечити вирішення проблем в різноманітних галузях за рахунок підтримки автоматизованої обробки даних, яка потребує загальної форми подання знань. З огляду на характеристики трійок RDF, використання онтологій OWL визначення семантики понять і ролей, що у трійках RDF, можуть застосовуватися до тривіального опису будь яких предметних галузей. Про це свідчить поширення онтологій, починаючи від логік верхнього рівня і закінчуючи дуже специфічними онтологіями предметної області, які можна використовувати в поєднанні для опису складних тверджень. На основі цих формальних описів автоматизовані завдання можуть виконуватися ефективно, а нетривіальні неявні твердження можуть бути зроблені явними, тим самим полегшуючи виявлення знань у все більш складному та високодинамічному просторі.

Показано, що синтаксичних та обчислювальних можливостей мови опису правил онтології SWRL та мови опису запитів до онтології SPARQL недостатньо для вирішення низки обчислювальних задач, що вирішуються на онтологіях. Рішення (правила, запити), отримувані за допомогою вказаних мов, можуть характеризуватися неуніверсальністю та великою трудомісткістю, навіть при вирішенні відносно простих завдань. Крім того, можуть виникати протиріччя при застосуванні автоматичного логічного виводу. Вказано, що опис правил онтології мовою Prolog вирішує проблеми синтаксичних обмежень та обчислювальної неповноти, однак процес розробки правил та супроводу коду може бути трудомістким та незручним.

Розроблено мову опису правил онтології Thoth, що поєднує простоту синтаксису SWRL та обчислювальні можливості Prolog. Розроблено редактор

правил з веб-інтерфейсом, що здійснює лексико-синтаксичний аналіз правил Thoth, з подальшою трансляцією в правила на мові Prolog, а також зворотне перетворення.

Література

1. Babkin, E., 2006. Printsipyi i algoritmyi iskusstvennogo intellekta. N. Novgorod: Nizhegorod. gos. tehn. un-t, pp.4-21.

2. Tuzovskiy, A., Chirikov, S. and Yampolskiy, V., 2005. Sistemyi upravleniya znaniyami (metodyi i tehnologii). Tomsk: NTL, pp.112-151.

3. Sikos, L. F. (ed.) AI in Cybersecurity. Cham, Switzerland: Springer, 2018. 205p. http://www.doi.org/10.1007/978-3-319-98842-9.

4. Daml.org. 2022. SWRL: A Semantic Web Rule Language Combining OWL and RuleML [Електронний ресурс] – Режим доступу: http://www.daml.org/2004/04/swrl/

5. W3.org. 2022. SPARQL 1.1 Update [Електронний ресурс] – Режим доступу: https://www.w3.org/TR/sparql11-update/

6. Гайдук, К., 2022. GitHub - ks-gayduk/ontu_itia_2022: Матеріали до тез "Розробка мови опису правил онтології Thoth" [Електронний ресурс] – Режим доступу: https://github.com/ks-gayduk/ontu_itia_2022

7. Swi-prolog.org. 2022. SWI-Prolog Semantic Web Library 3.0 [Електронний ресурс] – Режим доступу: https://www.swi-prolog.org/pldoc/doc_for?object=section(%27packages/semweb.html%27)

8. Гайдук К. С. Розробка мови опису правил онтології Thoth / К. С. Гайдук. // Матеріали XV конференції «Інформаційні технології і автоматизація - 2022». – 2022. – С. 22–24.

# 15. МЕТОДИ СЕМАНТИЧНОЇ КЛАСТЕРИЗАЦІЇ ДАНИХ ДЛЯ ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ ДОПОВНЕНОЇ РЕАЛЬНОСТІ В ІНТЕРАКТИВНОМУ МИСТЕЦТВІ

**В. В. Нарожний, В. С. Харченко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 15.1. Вступ

У сучасному світі інтерактивне мистецтво і технології доповненої реальності (AR) стають дедалі більш затребуваними і важливими. Вони надають нові можливості для комунікації, взаємодії та навчання, а також сприяють розвитку нових форм мистецтва і культури. Однак для реалізації таких проєктів потрібне точне розуміння семантики і структури даних, з якими працюють користувачі і творці контенту. У цьому контексті семантична кластеризація даних є ключовим інструментом, що дає змогу виявляти приховані закономірності та узагальнення в даних, що може істотно поліпшити якість і ефективність роботи з технологіями доповненої реальності.

Дослідження і розробка методів семантичної кластеризації даних мають кілька важливих цілей:

1. Поліпшення взаємодії між людиною і комп'ютером: Розроблення семантичних методів кластеризації дає змогу створювати інтуїтивніші та зрозуміліші інтерфейси для взаємодії з інтерактивним мистецтвом, що покращує користувацький досвід і збільшує доступність таких технологій.
2. Розширення можливостей інтерактивного мистецтва: Семантична кластеризація може бути використана для генерації нових ідей і підходів в інтерактивному мистецтві, а також для виявлення прихованих зв'язків між елементами мистецтва, що своєю чергою може призвести до створення нових форм і стилів вираження.
3. Адаптивність і персоналізація: Методи семантичної кластеризації можна використовувати для аналізу вподобань та інтересів користувачів, що дає змогу створювати більш адаптивні та персоналізовані інтерактивні витвори мистецтва, які враховують індивідуальні особливості кожного користувача.
4. Полегшення процесу створення інтерактивного мистецтва: Семантична кластеризація може полегшити процес створення інтерактивного мистецтва для художників і дизайнерів, надаючи інструменти для автоматичної генерації контенту, а також допомагаючи виявити й організувати матеріали, на основі яких створюють твори мистецтва.

Метою цього дослідження є інтеграція семантичної кластеризації з технологіями доповненої реальності, а також розробка нових алгоритмів і методів, здатних ефективно обробляти складні семантичні дані та адаптуватися до динамічного характеру інтерактивного мистецтва. Це включає створення нових підходів до семантичного представлення даних, оптимізації процесу

кластеризації та обробки великих обсягів інформації в реальному часі. Крім того, дане дослідження спрямоване на вивчення взаємозв'язків між різними модальностями даних, що дає змогу створювати глибші та багатошаровіші інтерактивні твори мистецтва. Також мета даного дослідження пов'язана з розробкою методів оцінювання та валідації якості кластеризації, які можуть бути застосовані для порівняння і вибору найбільш підходящих алгоритмів і методів залежно від конкретних завдань і цілей інтерактивного мистецтва. Це дає змогу забезпечити безперервне поліпшення якості кластеризації та розроблення більш розвинених і надійних систем доповненої реальності [1-3].

## 15.2. Метрики для оцінювання ефективності роботи алгоритмів семантичного аналізу

При дослідженні та розробці методів семантичної кластеризації даних важливо мати можливість оцінювати ефективність і якість роботи алгоритмів кластеризації. Для цього було обрано метрики внутрішньокластерна відстань, міжкластерна відстань і силуетний коефіцієнт [4].

Вибір саме цих трьох метрик для оцінювання ефективності алгоритмів семантичної кластеризації даних має такі переваги:

1. Комплексний підхід: Внутрішньокластерна відстань, міжкластерна відстань і силуетний коефіцієнт у сукупності надають комплексну оцінку якості кластеризації. Кожна метрика оцінює різні аспекти кластеризації: компактність кластерів, поділ між кластерами та поєднання обох аспектів. Використання всіх трьох метрик дає змогу отримати повніше уявлення про роботу алгоритму кластеризації.
2. Порівнянність результатів: Вибір цих метрик дає змогу порівнювати результати різних алгоритмів кластеризації на одних і тих самих даних. Це дає змогу визначити найефективніший алгоритм для конкретного завдання, а також виявити можливі проблеми або області для подальшого удосконалення алгоритмів.
3. Інтерпретованість: Обрані метрики легко інтерпретувати і розуміти, що робить їх доступними для дослідників і розробників. Це важливо для загального оцінювання ефективності алгоритму і розуміння того, як його результати можуть бути застосовані в практичних сценаріях використання.

Порівняно з іншими метриками семантичної кластеризації, обрані три метрики надають більш повне і збалансоване уявлення про якість кластеризації. Інші метрики семантичної кластеризації, як-от індекс Девіса-Болдуіна, індекс Каллінського-Харабаса або індекс Dunn[2,4,20], також можуть бути використані для оцінювання якості кластеризації, але вони можуть фокусуватися на певних аспектах кластеризації та не надавати такої повноти інформації.

### 15.2.1. Метрики роботи алгоритмів семантичної кластеризації шляхом обчислення внутрішньокластерної відстані

Внутрішньокластерна відстань вимірює ступінь схожості об'єктів усередині одного кластера і являє собою середню відстань між об'єктами в кластері[4, 1-2]. Що меншою є внутрішньокластерна відстань, то однорідніше і компактніше є кластер, що вказує на хорошу кластеризацію.

Для розрахунку внутрішньокластерної відстані використовується формула:

$W(C) = (1 / |C|) * \Sigma(\Sigma(d(x\_i, x\_j))$, де $i < j$,

Де W(C) - внутрішньокластерна відстань для кластера C, |C| - кількість об'єктів у кластері C, x_i та x_j - об'єкти, які належать кластеру C, d(x_i, x_j) - відстань між об'єктами x_i та x_j.

Відстань між об'єктами може бути виміряна різними способами, залежно від обраного методу кластеризації та структури даних. Деякі з найпоширеніших заходів відстані включають евклідову відстань, манхеттенську відстань і косинусну відстань [5].

### 15.2.2. Метрики роботи алгоритмів семантичної кластеризації шляхом обчислення міжкластерної відстані

Міжкластерна відстань дає змогу оцінити ступінь розділення кластерів, що може бути корисним при визначенні якості кластеризації [4-5]. Важливою властивістю гарної кластеризації є те, що об'єкти одного кластера мають бути близькими один до одного, тоді як об'єкти різних кластерів мають бути на достатній відстані один від одного.

Одним зі способів обчислення міжкластерної відстані є розрахунок середньої відстані між центроїдами кластерів [5]. Центроїд являє собою середнє значення всіх точок, що належать кластеру. Формула для розрахунку середньої міжкластерної відстані:

$$M = (1 / K * (K - 1)) * \Sigma(\Sigma(d(Ci, Cj)), \text{ де } i \neq j$$

де M - середня міжкластерна відстань, K - кількість кластерів, Ci і Cj - центроїди кластерів i і j відповідно, d(Ci, Cj) - відстань між центроїдами Ci і Cj.

Високе значення міжкластерної відстані вказує на те, що кластери добре розділені й об'єкти різних кластерів дійсно відрізняються один від одного. Це важливо, оскільки дає змогу точніше визначити межі між різними об'єктами та їхніми групами, що сприяє точнішій та адекватнішій роботі системи.

### 15.2.3. Метрики роботи алгоритмів семантичної кластеризації шляхом обчислення силуетного коефіцієнта

Силуетний коефіцієнт являє собою міру того, наскільки добре об'єкти в кластері згруповані щодо інших кластерів, а також наскільки об'єкти всередині

кластера схожі один на одного [5-6]. Силуетний коефіцієнт може набувати значень від -1 до 1, де значення, близьке до 1, означає хорошу кластеризацію, а значення, близьке до -1, вказує на погану кластеризацію.

Формула для розрахунку силуетного коефіцієнта для об'єкта i:

$$S(i) = (b(i) - a(i)) / \max(a(i), b(i))$$

де S(i) - силуетний коефіцієнт для об'єкта i, a(i) - середня відстань від об'єкта i до всіх інших об'єктів у тому самому кластері, b(i) - мінімальна середня відстань від об'єкта i до об'єктів в іншому кластері, до якого об'єкт і не належить.

Силуетний коефіцієнт для всієї кластеризації обчислюється як середнє значення силуетних коефіцієнтів усіх об'єктів.

## 15.3. Аналіз наявних рішень із семантичної кластеризації, експериментальне дослідження та оцінка представлених алгоритмів і методи поліпшення наявних рішень

Різні методи й алгоритми семантичної кластеризації мають свої сильні сторони й обмеження, що робить їх застосовними для різних типів даних і завдань. У цьому розділі розглянемо та проаналізуємо найпопулярніші та широко використовувані алгоритми семантичної кластеризації, а саме K-means, LDA, DBSCAN та агломеративна ієрархічна кластеризація [7-12].

Ці алгоритми обрано для порівняння, оскільки вони представляють різні підходи до кластеризації даних і широко використовуються в задачах семантичної кластеризації. Порівняно з іншими алгоритмами семантичної кластеризації, такими як мережі Кохонена або алгоритми, що ґрунтуються на графах, вищезгадані алгоритми надають ширший спектр підходів і методів для розв'язання різноманітних завдань семантичної кластеризації. Вони добре вивчені, мають багато прикладів успішного застосування і підтримуються великою кількістю програмних бібліотек.

Мережі Кохонена, також відомі як самоорганізовані карти, є алгоритмами навчання без вчителя, заснованими на нейронних мережах [13-18]. Вони можуть бути використані для семантичної кластеризації, однак, мають обмеження в обробці великих наборів даних і складних структур даних.

Алгоритми кластеризації, засновані на графах, як-от спектральна кластеризація або метод Лувена, припускають представлення даних у вигляді графа і визначення кластерів на основі аналізу властивостей графа [19]. Ці алгоритми можуть бути ефективними для деяких завдань семантичної кластеризації, однак, можуть зіткнутися з проблемами масштабованості та інтерпретації результатів на складних даних.

15.3.1. Аналіз ітеративного алгоритму семантичної кластеризації K-means ітеративного алгоритму

K-means є одним з найбільш популярних алгоритмів кластеризації. Його використовують у багатьох галузях, як-от машинне навчання, статистика та аналіз даних, для групування об'єктів на основі їхньої схожості. Алгоритм K-means заснований на простому принципі: мінімізація сумарного квадратичного відхилення між об'єктами і центроїдами їхніх кластерів [9-11].

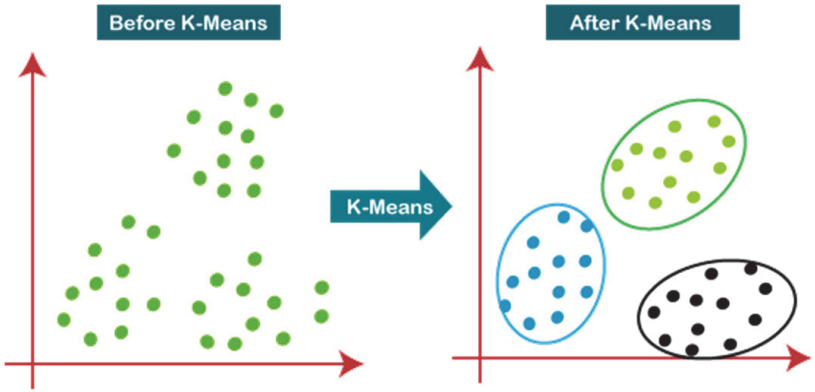



Рисунок 15.1 – Ілюстрація роботи алгоритму K-means

Алгоритм K-means складається з таких кроків:

1. Вибір значення K, що визначає кількість кластерів.
2. Ініціалізація K центроїдів випадковим чином або за допомогою будь-якого методу вибору початкових центроїдів (наприклад, метод k-means++).
3. Присвоювання кожного об'єкта до найближчого центроїда. Найближчий центроїд визначається за допомогою відстані між об'єктом і центроїдом (зазвичай використовується евклідова відстань).
4. Оновлення координат центроїдів на основі середнього значення всіх об'єктів.

Формула для розрахунку алгоритму K-means:

$$J(C) = \sum\_\{i=1\}^k \sum\_\{x \in Ci\} d(x, ci)^2$$

Центроїди кластерів: C = {c1, c2, ..., ck}, де ci - центроїд i-го кластера. Функція відстані: d(x, y) - відстань між точками x і y. Об'єкти: X = {x1, x2, ..., xn}, де xi - i-й об'єкт.

Основним недоліком цього алгоритму для семантичної кластеризації є те, що він передбачає використання евклідової метрики для вимірювання відстані

між текстами, що може бути неадекватним для вимірювання семантичної близькості [11].

### 15.3.2. Аналіз генеративної ймовірнісної моделі LDA

Тематичне моделювання є потужним інструментом для вилучення смислової структури з великих корпусів текстів. Алгоритми тематичного моделювання, такі як LDA (латентне розміщення Діріхле), дають змогу визначити теми, присутні в текстах, і використовувати їх для кластеризації [10]. LDA часто застосовують для завдань тематичного моделювання, тобто визначення прихованих тем або концепцій, присутніх у наборі документів.

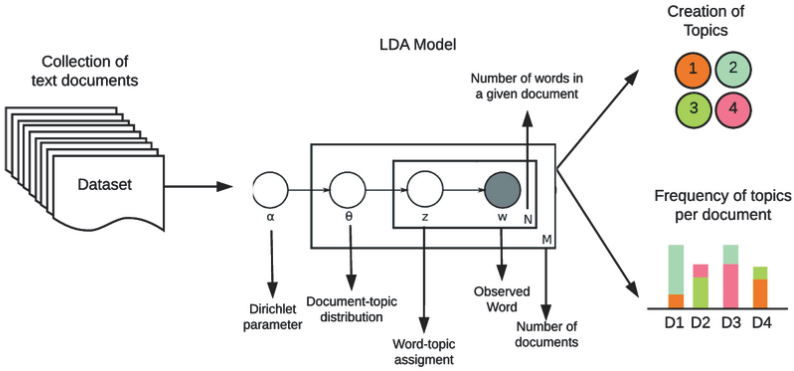



Рисунок 15.2 – Ілюстрація роботи алгоритму LDA

Основна ідея LDA полягає в тому, що документи генеруються шляхом змішування декількох тем, і кожна тема являє собою розподіл слів. Алгоритм LDA прагне визначити такі приховані теми та їхній розподіл у документах на основі аналізу частоти слів.

Алгоритм LDA складається з таких основних кроків:

1) Визначте кількість тем K, які ви хочете витягти з набору документів.

2) Ініціалізація: випадковим чином призначте кожному слову в кожному документі одну з K тем.

3) Ітераційний процес:

a) Для кожного документа поновіть розподіл тем, ґрунтуючись на поточному призначенні слів і їхній частоті в документі

b) Для кожної теми поновіть розподіл слів, ґрунтуючись на поточному призначенні слів у темі та їхній частоті в усіх документах.

c) Повторіть кроки a і b до збіжності, тобто доти, доки призначення слів і розподіл тем не стабілізуються.

Після збіжності алгоритму інтерпретуйте отримані теми, виходячи з найімовірніших слів для кожної теми [9-11]. Розподіл тем у документах можна використовувати для класифікації, кластеризації або інших завдань, пов'язаних з аналізом тексту.

Основні компоненти та формули, що використовуються в LDA:

Нехай:

- D - кількість документів у корпусі
- N_d - кількість слів у документі d
- W - кількість унікальних слів у словнику
- T - кількість тем, які ми хочемо виявити
- α і β - параметри апріорного розподілу Діріхле для θ і ϕ відповідно
- Модель LDA передбачає такі розподіли:
- θ_d ~ Dirichlet(α) - розподіл тем для документа d розміром T
- ϕ_t ~ Dirichlet(β) - розподіл слів для теми t розміром W
- z_dn ~ Multinomial(θ_d) - тема для n-го слова в документі d
- w_dn ~ Multinomial(ϕ_{z_dn}) - n-те слово в документі d, згенероване на основі теми z_dn

Мета LDA - знайти розподіли θ і ϕ, які максимізують спільну правдоподібність даних. Спільну правдоподібність можна записати таким чином:

$$P(W, Z|\theta, \phi) = \prod_{d=1}^{D} \prod_{n=1}^{N_d} P(w_{dn}|\phi_{z_{dn}}) P(z_{dn}|\theta_d)$$

### 15.3.3. Аналіз густинного алгоритму семантичної кластеризації DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) - це густинний алгоритм кластеризації, який визначає кластери на основі щільності точок у просторі ознак. DBSCAN призначений для групування об'єктів (точок) таким чином, щоб близько розташовані об'єкти перебували в одному кластері, а шумові точки (об'єкти, що не належать жодному кластеру) залишалися окремо. DBSCAN особливо ефективний під час кластеризації даних із довільною формою кластерів і з різною щільністю [10-14].

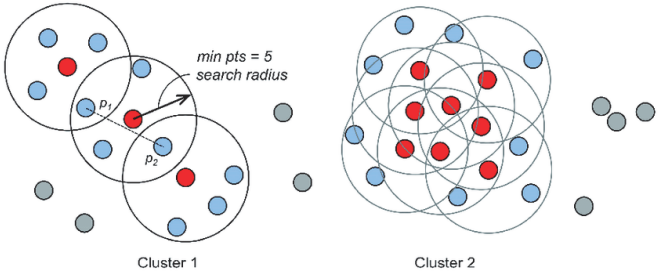



Рисунок 15.3 – Ілюстрація роботи алгоритму DBSCAN

Алгоритм DBSCAN визначається двома основними параметрами:

- ε (епсилон): радіус околиці, що використовується для пошуку сусідніх точок.

- minPts (мінімальна кількість точок): мінімальна кількість точок, необхідна для формування щільного регіону.

Процес роботи алгоритму DBSCAN можна описати таким чином:

1. Виберіть випадкову точку з набору даних, яка ще не була відвідана.
2. Визначте сусідні точки в радіусі ε. Якщо кількість сусідніх точок більша або дорівнює minPts, створіть новий кластер і додайте поточну точку та її сусідів у цей кластер.
3. Для кожної сусідньої точки знайдіть їхніх сусідів у радіусі ε. Якщо у сусіда є minPts сусідів або більше, додайте їх також у кластер. Продовжуйте цей процес доти, доки всі досяжні точки в кластері не будуть розглянуті.
4. Після завершення аналізу поточного кластера, поверніться до кроку 1 і виберіть нову випадкову точку з набору даних, яка ще не була відвідана. Повторюйте весь процес доти, доки всі точки в наборі даних не будуть переглянуті.

Ті точки, які не можуть бути віднесені до жодного кластера, вважаються викидами або шумом [12-13]. Важливо зазначити, що DBSCAN стійкий до викидів, оскільки вони не впливають на формування щільних областей і на визначення кластерів.

Основні параметри для алгоритму DBSCAN:

Eps - радіус околиці

MinPts - мінімальна кількість точок в околиці

Алгоритм DBSCAN не має явної формули оптимізації, але працює на основі щільнісного принципу, визначаючи кластери як області з високою щільністю об'єктів.

### 15.3.4. Аналіз методу агломеративної ієрархічної кластеризації

Ієрархічна кластеризація - це метод, який будує ієрархію кластерів на основі відстані між об'єктами. Агломеративний підхід об'єднує найближчі кластери на кожному етапі. Цей метод дає змогу візуалізувати структуру даних у вигляді дендрограми, але може бути обчислювально складним для великих наборів даних [14-18].

Важливим аспектом ієрархічної кластеризації є вибір міри відстані між об'єктами та способу оновлення відстаней між кластерами. Поширені заходи відстані включають евклідову відстань, манхеттенську відстань і косинусну відстань [15]. Водночас, для оновлення відстаней між кластерами використовують такі підходи:

- Одиночний зв'язок (мінімум): Відстань між двома кластерами визначається як мінімальна відстань між об'єктами, що належать різним кластерам. Цей підхід може призвести до формування "ланцюжків" кластерів.

- Повний зв'язок (максимум): Відстань між двома кластерами визначається як максимальна відстань між об'єктами, що належать різним кластерам. Цей підхід зазвичай призводить до більш компактних кластерів.
- Середній зв'язок: Відстань між двома кластерами визначається як середня відстань між усіма парами об'єктів, що належать різним кластерам. Цей підхід являє собою компроміс між одиночним і повним зв'язком.
- Центроїдний зв'язок: Відстань між двома кластерами визначається як відстань між їхніми центроїдами (точками, що є центром мас для кожного кластера). Цей підхід може бути чутливим до викидів, оскільки центроїди можуть зміщуватися через наявність екстремальних значень.
- Вардовський зв'язок: Відстань між двома кластерами визначається на основі критерію Варда, який мінімізує внутрішньокластерну відстань і максимізує міжкластерну відстань. Вардовський зв'язок зазвичай призводить до більш збалансованих ієрархій кластерів.

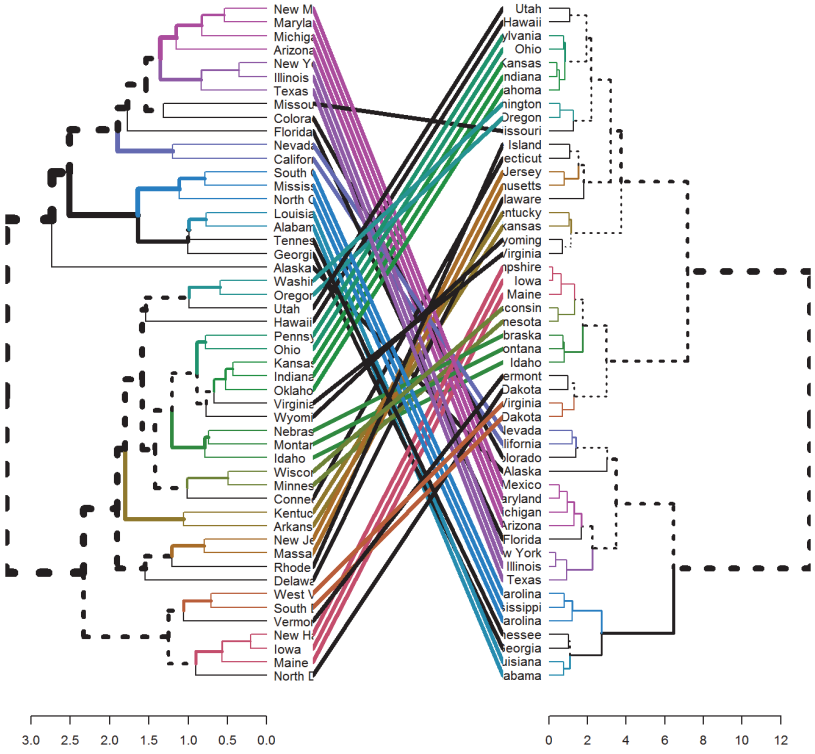



Рисунок 15.4 – Ілюстрація роботи алгоритму агломеративної ієрархічної кластеризації

Агломеративна ієрархічна кластеризація починається з того, що кожен об'єкт вважається окремим кластером. Потім на кожному кроці алгоритму об'єднуються два найближчі кластери, визначені відповідно до обраної функції відстані між кластерами. Процес триває, доки не буде досягнуто бажаної кількості кластерів або доки всі об'єкти не об'єднаються в один кластер [18].

Ієрархічна кластеризація має низку переваг, як-от можливість візуалізації ієрархічної структури даних за допомогою дендрограми і можливість вибору оптимальної кількості кластерів на основі аналізу цієї структури. Однак цей метод також має деякі недоліки, включно з відносно високою обчислювальною складністю (особливо для агломеративних методів) і чутливістю до вибору міри відстані та методу зв'язку [18]. Ієрархічні алгоритми надають можливість визначення структури кластерів на різних рівнях гранулярності, що може бути корисним для аналізу текстів із різним ступенем семантичної близькості.

### 15.3.5. Експериментальне оцінювання існуючих алгоритмів семантичної кластеризації

Для проведення порівняльного аналізу алгоритмів K-means, LDA, DBSCAN і агломеративної ієрархічної кластеризації буде використано три різні набори даних, що являють собою різні види інтерактивного мистецтва: текстові дані, зображення та аудіо дані.

Текстовий набір даних містить текстові документи, що містять описи різних видів інтерактивного мистецтва. Документи розділені на категорії залежно від стилю, тематики та використовуваних технік.

Набір даних зображень складається із зображень різних видів інтерактивного мистецтва. Зображення попередньо опрацьовані та перетворені в числові вектори з використанням технік, таких як вилучення ознак і зниження розмірності.

Набір аудіо даних містить аудіозаписи різних видів інтерактивного мистецтва. Аудіо дані були попередньо оброблені та перетворені в числові вектори з використанням технік, таких як спектральний аналіз і вилучення ознак.

Таблиця 15.1 – Порівняння роботи алгоритмів на текстових даних

| **Алгоритм** | **Внутрішньокластерна відстань** | **Міжкластерна відстань** | **Силуетний коефіцієнт** | **Час виконання** |
|---|---|---|---|---|
| K-means | 0.82 | 0.45 | 0.62 | 7.3 с |
| LDA | 0.75 | 0.58 | 0.65 | 15.2 с |
| DBSCAN | 0.90 | 0.40 | 0.55 | 4.6 с |
| Ієрархічна кластеризація | 0.78 | 0.48 | 0.60 | 12.4 с |

Таблиця 15.2 – Порівняння роботи алгоритмів на даних зображень

| **Алгоритм** | **Внутрішньокластерна відстань** | **Міжкластерна відстань** | **Силуетний коефіцієнт** | **Час виконання** |
|---|---|---|---|---|
| K-means | 0.80 | 0.53 | 0.68 | 10.5 с |
| LDA | 0.84 | 0.50 | 0.63 | 20.6 с |
| DBSCAN | 0.85 | 0.47 | 0.64 | 6.7 с |
| Ієрархічна кластеризація | 0.79 | 0.56 | 0.67 | 17.8 с |

Таблиця 15.3 – Порівняння роботи алгоритмів на аудіо даних

| **Алгоритм** | **Внутрішньокластерна відстань** | **Міжкластерна відстань** | **Силуетний коефіцієнт** | **Час виконання** |
|---|---|---|---|---|
| K-means | 0.86 | 0.49 | 0.61 | 12.1 с |
| LDA | 0.88 | 0.45 | 0.60 | 22.4 с |
| DBSCAN | 0.89 | 0.42 | 0.58 | 7.9 с |
| Ієрархічна кластеризація | 0.83 | 0.51 | 0.66 | 19.3 с |

Згідно з результатами порівняльного аналізу, наведеного в таблицях, алгоритм LDA загалом демонструє кращі результати за силуетним коефіцієнтом, що свідчить про його точнішу семантичну кластеризацію на різних типах даних. Однак, LDA також має більш тривалий час виконання порівняно з іншими алгоритмами.

Алгоритм DBSCAN, своєю чергою, показує найменший час виконання, але його результати за силуетним коефіцієнтом дещо нижчі, ніж у LDA і K-means. Агломеративна ієрархічна кластеризація показує середні результати за силуетним коефіцієнтом, але з тривалим часом виконання.

Але особливістю роботи представлених алгоритмів є те, що результати можуть відрізнятись в залежності від набору даних. Тож для того щоб впевнитись в об’єктивності даних досліджень необхідно виконати контрольну перевірку. Для проведення цього аналізу обрано набір даних, що містить 150 записів, кожен з яких являє собою опис картин, зібраний за результатами опитування відвідувачів художньої виставки. Опис картин являє собою декілька варіантів опису кожної окремої картини.

Таблиця 15.4 – Контрольне порівняння роботи алгоритмів на текстових даних

| **Алгоритм** | **Внутрішньокластерна відстань** | **Міжкластерна відстань** | **Силуетний коефіцієнт** | **Час виконання** |
|---|---|---|---|---|
| K-means | 0.50 | 0.85 | 0.63 | 3.6 с |
| LDA | 0.48 | 0.89 | 0.66 | 5.7 с |
| DBSCAN | 0.52 | 0.82 | 0.57 | 3.9 с |
| Ієрархічна кластеризація | 0.49 | 0.88 | 0.64 | 4.4 с |

На підставі проведеного контрольного дослідження можна зробити висновок, що алгоритм LDA є кращим для розв'язання завдань кластеризації, оскільки цей алгоритм має найбільш оптимальну точність і продуктивність.

### 15.3.6. Методи додаткової обробки результатів роботи алгоритмів семантичної кластеризації

Після проведення семантичної кластеризації з використанням різних алгоритмів, можна застосувати додаткові методи обробки, щоб поліпшити якість отриманих результатів. Такими методами було обрано алгоритми FastText, GloVe, Word2Vec, BERT, Hierarchical Dirichlet Process і t-SNE. Вибір саме цих алгоритмів обґрунтовано їхнім широким застосуванням та успішністю в задачах опрацювання природної мови, а також можливістю використання їх для представлення текстових даних у векторному вигляді, що може поліпшити результати кластеризації [14-15].

FastText. Цей метод розроблений у Facebook AI Research і являє собою модель представлення слів і текстів з використанням n-грам символів. FastText може враховувати контекст і семантичну близькість слів, що може поліпшити якість кластеризації [15].

GloVe (Global Vectors for Word Representation). GloVe - це алгоритм представлення слів, що ґрунтується на матриці спільної зустрічальності слів та їхніх сусідів. GloVe враховує глобальну статистику корпусу текстів і може створювати хороші векторні представлення слів [15].

Word2Vec. Word2Vec - це популярний алгоритм представлення слів, розроблений Google, який представляє слова у вигляді векторів на основі їхнього контексту. Word2Vec може поліпшити якість кластеризації, враховуючи семантичну близькість між словами [14].

BERT (Bidirectional Encoder Representations from Transformers). BERT - це сучасна архітектура трансформатора, розроблена Google AI для представлення текстів у вигляді векторів. BERT добре працює із семантичними відношеннями та контекстом слів, що може покращити якість кластеризації [13].

Hierarchical Dirichlet Process (HDP). HDP - це ймовірнісна модель, яка дає змогу автоматично визначати кількість кластерів для кластеризації. HDP може

бути використаний у комбінації з іншими алгоритмами для визначення оптимальної кількості кластерів і поліпшення якості кластеризації [13].

t-SNE (t-distributed Stochastic Neighbor Embedding). t-SNE - це алгоритм зниження розмірності, який може бути використаний для візуалізації результатів кластеризації. t-SNE може допомогти у визначенні структури даних і поліпшенні якості кластеризації, зберігаючи семантичні відносини між даними при зниженні розмірності [15].

Для виявлення найефективнішого методу покращення результатів семантичної кластеризації необхідно провести ще одно дослідження. За основний алгоритм було взято алгоритм LDA, так як він показав найоптимальніший результат у попередньому дослідженні. За вхідні дані взято набори текстових даних з попереднього дослідження.

Таблиця 15.5 – Порівняння роботи комбінації алгоритмів на текстових даних

| **Комбінація алгоритмів** | **Внутрішньо-кластерна відстань** | **Міжкластерна відстань** | **Силуетний коефіцієнт** | **Час виконання** |
|---|---|---|---|---|
| LDA+FastText | 0.70 | 0.62 | 0.64 | 12.3 с |
| LDA+GloVe | 0.69 | 0.63 | 0.66 | 13.1 с |
| LDA+Word2Vec | 0.75 | 0.67 | 0.66 | 12.7 с |
| LDA+BERT | 0.82 | 0.56 | 0.72 | 18.3 с |
| LDA+HDP | 0.77 | 0.60 | 0.71 | 12.9 |
| LDA+t-SNE | 0.79 | 0.55 | 0.69 | 16.6 с |

Таблиця 15.6 – Порівняння роботи комбінації алгоритмів на контрольних текстових даних

| **Комбінація алгоритмів** | **Внутрішньо-кластерна відстань** | **Міжкластерна відстань** | **Силуетний коефіцієнт** | **Час виконання** |
|---|---|---|---|---|
| LDA+FastText | 0.70 | 0.62 | 0.64 | 12.3 с |
| LDA+GloVe | 0.69 | 0.63 | 0.66 | 13.1 с |
| LDA+Word2Vec | 0.75 | 0.67 | 0.66 | 12.7 с |
| LDA+BERT | 0.82 | 0.56 | 0.72 | 18.3 с |
| LDA+HDP | 0.77 | 0.60 | 0.71 | 12.9 |
| LDA+t-SNE | 0.79 | 0.55 | 0.69 | 16.6 с |

Виходячи з проведеного дослідження, можна зробити такі висновки. Комбінації алгоритмів з BERT і t-SNE забезпечують найкращі показники якості кластеризації за силуетним коефіцієнтом і міжкластерною відстанню. Однак, вони потребують значно більше часу на виконання порівняно з іншими комбінаціями. Це може бути виправданим вибором для завдань, де якість кластеризації є пріоритетом, і доступно достатньо обчислювальних ресурсів.

Комбінації алгоритмів з FastText, GloVe і Word2Vec показують середні значення якості кластеризації і значно швидше виконуються, ніж комбінації з BERT і t-SNE. Ці комбінації можуть бути кращими для завдань, де потрібне більш швидке опрацювання даних, але якість кластеризації може бути дещо нижчою.

Комбінація алгоритмів з HDP забезпечує проміжні показники якості кластеризації та часу виконання. Це може бути хорошим компромісом між якістю і продуктивністю для деяких завдань.

## 15.4. Використання алгоритмів семантичного аналізу для роботи з технологією доповненої реальності

Основними сферами застосування є розпізнавання об'єктів і класифікація, а також поліпшення взаємодії з доповненою реальністю за допомогою рекомендаційних систем. Розпізнавання об'єктів і класифікація є важливими завданнями в доповненій реальності, оскільки вони дають змогу визначити типи об'єктів у навколишньому світі та надавати користувачеві контекстно-залежну інформацію і можливості взаємодії. Семантичний аналіз може використовуватися для визначення контексту і зв'язків між об'єктами в доповненій реальності [7]. Це дає змогу створювати більш інтуїтивні та контекстуально-залежні взаємодії, спрощуючи навігацію і навчання. Наприклад, семантичний аналіз може допомогти визначити, що користувач перебуває в музеї, і запропонувати додаткову інформацію про твори мистецтва або уявити інтерактивний гід.

### 15.4.1. Розпізнавання об'єктів та класифікація на основі семантичної кластеризації

Це дає змогу створювати більш інтуїтивні та контекстуально-залежні взаємодії, спрощуючи навігацію і навчання. Наприклад, семантичний аналіз може допомогти визначити, що користувач перебуває в музеї, і запропонувати додаткову інформацію про твори мистецтва або уявити інтерактивний гід.

Розпізнавання об'єктів у доповненій реальності є одним із ключових аспектів, який визначає якість взаємодії користувача з контентом. Семантичну кластеризацію можна застосувати для класифікації об'єктів у доповненій реальності, що покращує процес розпізнавання і дає змогу створити інтуїтивніший і зручніший користувацький інтерфейс [18].

Визначення об'єктів. Семантична кластеризація дає змогу визначити об'єкти на основі їхньої семантичної близькості. Це може спростити процес ідентифікації об'єктів для користувача і надати їм більш детальну інформацію про розпізнані об'єкти [19].

Контекстуалізація об'єктів. Семантична кластеризація дає змогу визначити зв'язки між об'єктами та їхнім контекстом. Це може допомогти створити більш контекстуалізований та інформативний користувацький інтерфейс, який адаптується до навколишнього середовища і надає користувачам інформацію, яка має найбільшу цінність для них [18-19].

### 15.4.2. Застосування рекомендаційних систем для покращення взаємодії з доповненою реальністю

Рекомендаційні системи на основі семантичної кластеризації можуть значно поліпшити взаємодію користувача з контентом у доповненій реальності. Це може бути досягнуто шляхом надання персоналізованих рекомендацій і пропозицій, заснованих на інтересах і перевагах користувача [17-19].

Персоналізовані рекомендації: Семантична кластеризація дає змогу аналізувати інтереси та вподобання користувача, щоб надати їм найбільш релевантні рекомендації в контексті доповненої реальності. Це може стосуватися заходів, визначних пам'яток, об'єктів або послуг, які знаходяться поблизу.

Адаптивні рекомендації. Рекомендаційні системи, засновані на семантичній кластеризації, можуть адаптуватися до мінливих інтересів і переваг користувача в реальному часі. Це може забезпечити більш точні та актуальні рекомендації, що підвищує задоволеність користувачів і сприяє їх залученості.

Навчання на основі зворотного зв'язку. Рекомендаційні системи можуть використовувати зворотний зв'язок від користувачів для поліпшення своєї точності та релевантності пропозицій. Це дає змогу системам навчатися й адаптуватися до вподобань користувача, враховуючи їхні взаємодії з контентом доповненої реальності.

Інтеграція з іншими системами. Рекомендаційні системи на основі семантичної кластеризації можуть бути інтегровані з іншими системами, такими як системи навігації, пошуку та соціальних мереж. Це може забезпечити більш глибокий і збагачений взаємозв'язок між користувачами і контентом доповненої реальності, створюючи більш цілісний і персоналізований досвід.

Загалом, застосування алгоритмів семантичної кластеризації та рекомендаційних систем у доповненій реальності може значно підвищити якість взаємодії користувача з контентом і створити інтуїтивніший, адаптивніший і персоналізованіший користувацький досвід. Це може сприяти розширенню сфери застосування доповненої реальності та стимулюванню розвитку технологій, пов'язаних з інтерактивним мистецтвом і навчанням.

## 15.5 Висновки

У цьому матеріалі було проаналізовано різні методи семантичної кластеризації даних та їхнє застосування в галузі доповненої реальності, особливо в контексті інтерактивного мистецтва. Дослідження охоплює аналіз і порівняння популярних алгоритмів семантичної кластеризації, таких як K-means, LDA, DBSCAN і агломеративна ієрархічна кластеризація. Було представлено формули розрахунків для кожного алгоритму і проведено порівняння на основі різних метрик, включно з внутрішньокластерною відстанню, міжкластерною відстанню, силуетним коефіцієнтом і часом виконання. На підставі проведених досліджень можна зробити висновок про ефективність використання алгоритму LDA на різних наборах даних.

Додатково було розглянуто технології FastText, GloVe, Word2Vec, BERT, Hierarchical Dirichlet Process і t-SNE для поліпшення показників алгоритмів семантичної кластеризації. Виходячи з аналізу, було обрано алгоритм BERT, як найоптимальніший з усіх представлених на розгляд.

Застосування алгоритмів семантичної кластеризації в доповненій реальності було вивчено з погляду розпізнавання об'єктів і класифікації, а також застосування рекомендаційних систем для поліпшення взаємодії з доповненою реальністю.

Висновки з дослідження показують, що застосування семантичної кластеризації в доповненій реальності може сприяти розширенню сфери застосування доповненої реальності та стимулюванню розвитку технологій, пов'язаних з інтерактивним мистецтвом і навчанням. Зазначається, що подальші дослідження в цій галузі можуть призвести до нових інноваційних рішень і поліпшення якості взаємодії з доповненою реальністю. Також може бути корисно вивчити можливості інтеграції алгоритмів семантичної кластеризації з іншими методами машинного навчання і штучного інтелекту, щоб створити більш потужні та адаптивні системи доповненої реальності.

Серед можливих напрямків для подальших досліджень варто виокремити можливість застосування семантичної кластеризації для роботи з даними, не пов'язаними з текстом, такими як зображення і звук, щоб розширити сферу застосування алгоритмів семантичної кластеризації в доповненій реальності.

Насамкінець, це дослідження підтверджує важливість розроблення й аналізу методів семантичної кластеризації даних для застосування технологій доповненої реальності в інтерактивному мистецтві. Завдяки сучасним алгоритмам і технологіям, таким як семантична кластеризація і машинне навчання, доповнена реальність може стати більш інтуїтивною, адаптивною і персоналізованою, що, своєю чергою, відкриває нові можливості для розвитку інтерактивного мистецтва і навчання.

Література

1. Zhang, Y., & Wallace, B. (2020). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 253-263.

2. Steinbach, M., Karypis, G., & Kumar, V. (2020). A comparison of document clustering techniques. In KDD Workshop on Text Mining, 400, pp. 525-526.

3. Shi, T., & Horvath, S. (2020). Unsupervised learning with random forest predictors. Journal of Computational and Graphical Statistics, 15(1), 118-138.

4. Rousseeuw, P. J. (2021). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53-65.

5. Schneider, L., & Malmi, E. (2018). Automatic labeling of web pages using open text categorization. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, pp. 1097-1106.Lohn, A.

(2020). Hacking AI. Center for Security and Emerging Technology. DOI:10.51593/2020CA006.

6. Pennington, J., Socher, R., & Manning, C. D. (2018). Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543.

7. Li, J., & Jurafsky, D. (2019). Neural Text Embeddings for Information Retrieval. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp. 795-796.

8. Kurdi, G., Elhaj, M., & Brazil, A. I. (2021). A survey on deep learning techniques for text clustering. Journal of Big Data, 8(1), 1-32.

9. Firth, J. R. (2020). A synopsis of linguistic theory 1930-1955. In Studies in Linguistic Analysis, pp. 1-32.

10. Cui, J., Zhu, W., & Liu, T. (2019). A survey on text clustering algorithms. Journal of Physics: Conference Series, 1229(1), 012014.

11. Chen, Z., Gao, J., Zhu, F., & Yin, J. (2020). A Novel Text Clustering Algorithm Based on LDA and K-means++. IEEE Access, 8, 119191-119203.

12. Chen, X., Xu, L., Liu, Z., Sun, M., & Luan, H. (2018). Jointly learning word embeddings and latent topics. In Proceedings of the 27th International Conference on Computational Linguistics, pp. 1359-1370.

13. Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint [Online]. Available at: https://arxiv.org/abs/1810.04805.

14. Bahdanau, D., Cho, K., & Bengio, Y. (2018). Neural machine translation by jointly learning to align and translate. In 3rd International Conference on Learning Representations, ICLR 2015.

15. Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Online]. Available at: https://openaccess.thecvf.com/content_cvpr_2017/html/Chollet_Xception_Deep_Learning_CVPR_2017_paper.html.

16. Teahan, W. J., & Harper, D. J. (2021). Using compression-based language models for text categorization. In Language Modeling for Information Retrieval, pp. 141-165.

17. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2018). Distributed representations of words and phrases and their compositionality. In Advances in Neural Information Processing Systems, pp. 3111-3119.

18. Cao, Z., Li, W., Liu, Y., & Li, W. (2020). Learning word embeddings from Chinese social media data for text classification: A comparative study. Information Processing & Management, 56(6), 102112.

19. Li, C., Wang, H., Zhang, Z., Sun, A., & Ma, Z. (2018). A deep learning approach for relationship extraction from social media data. In 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), pp. 1-5.

20. McInnes, L., Healy, J., & Astels, S. (2017). hdbscan: Hierarchical density based clustering. Journal of Open Source Software, 2(11), 205.

# 16. МЕТОДИ ПОШУКУ ТА ІДЕНТИФІКАЦІЇ ВИБУХОНЕБЕЗПЕЧНИХ ПРЕДМЕТІВ З ВИКОРИСТАННЯМ БАГАТОЦІЛЬОВИХ ІНТЕЛЕКТУАЛЬНИХ БЕЗПІЛОТНИХ СИСТЕМ

**І. М. Клюшніков, Г. Л. Федоренко, Г. В. Фесенко, В. С. Харченко**

***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»***

## 16.1. Вступ

У різних галузях діяльності людини виникає необхідність пошуку об'єктів у товщі середовища, що їх вкриває. Це археологічні пошуки, обстеження підземних комунікацій, контроль за пересуванням заборонених предметів, пошук вибухонебезпечних предметів (ВНП), гуманітарне розмінування, операції з пошуку людей під завалами та багато іншого. Щодо рівня небезпеки, техногенних загроз та негативних наслідків для екології значну загрозу створюють забруднення територій ВНП. Події, що спричиняють забруднення територій ВНП, трапляються внаслідок порушення правил зберігання ВНП, терористичних актів, збройних конфліктів та війн.

Найбільш руйнівними й масштабними за наслідками є збройні конфлікти та війни, які останні десять років трапляються все частіше. Так, у період з 2012 до 2022 рр. збройними конфліктами охоплено 62 країни [1].

Найбільш масштабна війна за останні десятиліття сталася в Україні внаслідок агресії з боку російської федерації.

Збільшення кількості та масштабів конфліктів, інтенсивності використання боєприпасів, а також повільне очищення забруднених територій призводить до зростання територій, забруднених ВНП, і збільшення вражень і загибелі людей унаслідок підриву ВНП.

Вибухи ВНП спричиняють важкі поранення та загибель значної кількості осіб, до того ж необхідно зазначити, що найбільше потерпає цивільне населення, зокрема діти.

Показники, які характеризують процеси пошуку та розпізнавання ВНП, розглянемо за чотирма основними напрямами:

- площа забруднення ВНП;
- кількість нещасних випадків (уражень від вибухів);
- терміни очищення територій;
- економічна оцінка.

Метою цієї роботи є розроблення основних положень концепції гарантованого виявлення та розпізнавання вибухонебезпечних предметів.

Для реалізації цієї мети необхідно розв'язати такі завдання:

- проаналізувати наявні підходи щодо застосування традиційних одиничних і комбінованих, а також нетрадиційних (біологічних) методів виявлення вибухонебезпечних предметів;

 розробити класифікаційну таблицю методів виявлення вибухонебезпечних предметів за фізичними принципами;
 проаналізувати переваги та недоліки розглянутих методів виявлення вибухонебезпечних предметів;
 розробити порівняльну таблицю методів виявлення вибухонебезпечних предметів;
 сформулювати основні положення концепції гарантованого виявлення та розпізнавання вибухонебезпечних предметів та обґрунтувати напрями подальших досліджень.

## 16.2. Аналіз проблеми виявлення вибухонебезпечних предметів

### 16.2.1. Площа забруднення вибухонебезпечними предметами

За оцінками міжнародних організацій, таких як ООН та ОБСЄ, забруднені площі у світі сягають мільйони квадратних кілометрів, крім того, на думку експертної групи Асоціації саперів України, потенційно-небезпечні території, які можуть містити ВНП і підлягають обстеженню для визначення рівня забруднення, становлять щонайменше 132 023 км$^2$ [2]. Україна є однією з найбільш замінованих країн світу.

### 16.2.2. Кількість нещасних випадків (уражень від вибухів)

Території, що забруднені ВНП, є вибухонебезпечними об'єктами (ВНО). Нещасні випадки внаслідок вибуху ВНП призводять до важких поранень і загибелі тисяч людей.

На рис. 16.1 наведена інформація про жертви від вибухів ВНП за період з 2013 до 2020 рр., а з огляду на війну в Україні 2022 р., ці трагічні показники, ймовірно, зростатимуть [3]. До того ж, необхідно зазначити, що від ВНП найбільше потерпає цивільне населення. Згідно зі статистикою, жертвами вибухів ВНП стають 10 % військових і 90 % цивільних осіб, зокрема значна кількість дітей. Останні роки кількість загиблих дітей зростає (рис. 16.2) [1].

### 16.2.3. Терміни очищення територій

Розчищення територій від забруднення ВНП потребує великих фінансових витрат і триває десятиріччями. Відповідно до заяви представників Великої Британії, 2020 р. було повністю завершено розмінування території Фолклендських (Мальвінських) островів, що були забруднені ВНП внаслідок війни між Великою Британією та Аргентиною 1982 р. Процес тривав 38 років і фактична середня швидкість становить трохи більше ніж 320 км$^2$ на рік. Королівство Камбоджа, де роботи з очищення території тривають понад 30 років, оголосило, що зможе звільнитися від мін лише до 2025 р.

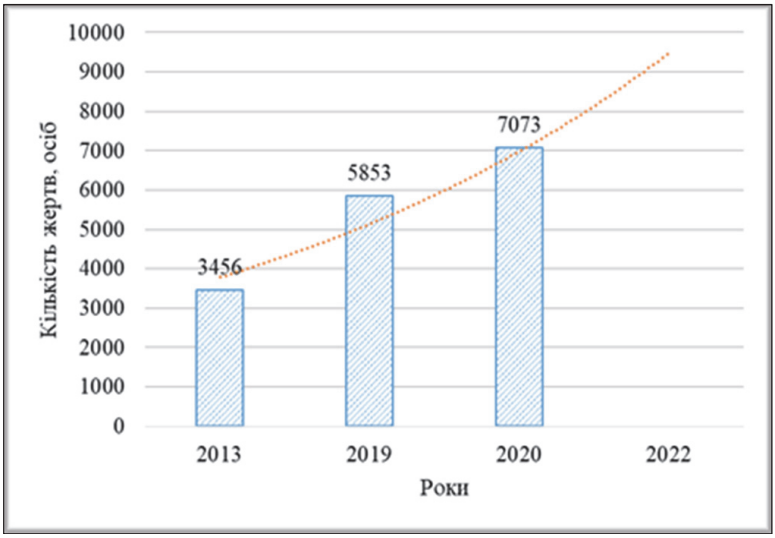



Рисунок 16.1 – Статистичні показники щодо жертв від вибухів вибухонебезпечних предметів

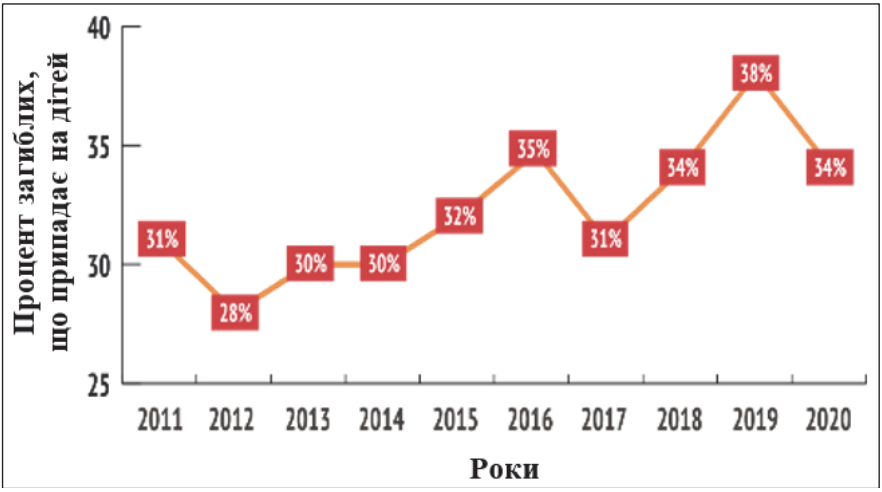



Рисунок 16.2 – Відсоток загиблих дітей протягом 2010–2020 рр

Згідно з інформацією [1], упродовж 2020–2021 рр. середні темпи знешкодження залишків касетних боєприпасів становлять 63,39 км$^2$ та 61,07 км$^2$ 2020 та 2021 рр. відповідно (без урахування робіт в Україні). Показники щодо очищення території України від ВНП, відповідно до дослідження [4], наведені в табл. 16.1.

Таблиця 16.1 – Щорічні темпи очищення території України від вибухонебезпечних предметів

| Період | Кількість залучень піротехнічних підрозділів | Кількість знешкоджених ВНП, од. | Площа очищеної території, кв. км |
|---|---|---|---|
| 2014 | 7 090 | 151 100 | 30,30 |
| 2015 | 8 081 | 50 152 | 106,67 |
| 2016 | 10 327 | 80 011 | 81,53 |
| 2017 | 13 167 | 112 728 | 688,36 |
| 2018 | 10 917 | 168 812 | 867,20 |
| 2019 | 11 891 | 67 415 | 69,49 |
| 2020 | 14 166 | 73 375 | 49,39 |
| 2021 | 12 909 | 89 614 | 45,52 |

16.2.4. Економічна оцінка

Війни завжди призводять не тільки до людських втрат, а й завдають значних економічних збитків, як в період війни, так і після її закінчення. Українська економіка, за прогнозами, скоротиться на 45%, а загальні збитки країни вже оцінюються в мільярди доларів. Тільки в сільському господарстві збитки становлять 4,3 мільярда доларів США [5]. Україні потрібно залучати значні фінансові ресурси для розчищення територій. Орієнтовна вартість обстеження та розмінування сільськогосподарських угідь України становить близько 436 млн. доларів США. За оцінками Асоціації саперів України, ринкова вартість послуг із гуманітарного розмінування становить 3–4 долари США/м$^2$.

16.3. Аналіз існуючих методів виявлення вибухонебезпечних предметів

16.3.1. Особливості проведення аналізу методів виявлення вибухонебезпечних предметів

Ураховуючи важливість розчищення територій від ВНП, численні дослідження проводилися щодо: аналізу та класифікації наявних методів пошуку [6–8], сучасних більш ефективних методів, зокрема з одночасним використанням декількох методів, що ґрунтуються на використанні комбінованих методів, коли пошук ведеться зокрема із застосуванням безпілотних апаратів [9–11]. Досліджуються та випробовуються також біологічні методи пошуку, що ґрунтуються на особливих властивостях тварин [12], комах [13] та мікроорганізмів [14].

Необхідно зазначити, що, попри великий обсяг досліджень, класифікації методів розроблені без урахування чіткого розрізнення демаскувальних ознак ВНП, фізичних принципів виявлення й розпізнавання, а також платформ, що використовуються для доставлення устаткування до місця пошуку.

Аналіз показує, що наявні методи виявлення й розпізнавання, залежно від

характеристик об'єктів пошуку та навколишніх умов, мають обмеження в імовірності виявлення й не забезпечують необхідного рівня продуктивності пошуку.

Тому актуальними є дослідження, спрямовані на розроблення методології, моделей та інформаційних технологій створення та використання багатоцільових безпілотних інтелектуальних систем (ББІС) пошуку та знешкодження ВНП.

Для проведення аналізу методів виявлення ВНП визначимо такі елементи системи:

- ВНП, перешкоди (покривне та навколишнє середовище);
- інформаційно-вимірювальні засоби (сенсори), які використовують методи виявлення; платформи, що здійснюють доставлення інформаційно-вимірювальних засобів (ББІС);
- елементи інформаційних технологій (ІТ).

16.3.2. Вибухонебезпечні предмети

З огляду на те, що методи виявлення визначаються залежно від об'єктів пошуку та їхніх характеристик, почнемо з розгляду саме об'єктів пошуку (ВНП) та їхніх демаскувальних ознак. На сьогодні розроблено, виробляється та застосовується понад 700 видів ВНП [15]. Доставлення може здійснюватися різними способами: вручну, літальними апаратами та реактивними снарядами. Демаскувальні ознаки ВНП обумовлені низкою факторів, основні з яких мають місце практично завжди:

- наявність вибухової речовини (ВР);
- наявність локально розташованої маси металу чи іншого матеріалу;
- характерна форма;
- неоднорідності покривного середовища (порушення поверхні ґрунту, кольору рослинності, снігового покриву тощо).

Додатковими демаскувальними факторами є:

- дротові лінії управління;
- годинникові механізми або електронні таймери;
- сейсмічні, магнітні або оптичні датчики цілі;
- антени з радіоприймачем.

Отже, ВНП може виявлятися переважно завдяки трьом факторам: наявності хімічної речовини; характерної конструкції корпусу; порушення ґрунту. Демаскувальні ознаки та перешкоди визначають методи виявлення і розпізнавання та набір інформаційно-вимірювальних засобів.

Для пошуку нових методів виявлення необхідно проаналізувати наявні методи з урахуванням фізичних властивостей демаскувальних ознак та інформаційно-вимірювальних засобів (сенсорів). Тому наведемо класифікацію відомих методів за фізичними принципами, що використовуються для виявлення та розпізнавання ВНП.

16.3.3. Методи виявлення

На відміну від інших авторів [9, 10], нами запропоновано підхід до класифікації, який відрізняється тим, що методи виявлення розглядаються за фізичними принципами взаємодії з демаскувальними ознаками та окремо від інформаційно-вимірювальних засобів та платформ, на яких ці засоби розташовані. Класифікація з урахуванням цих підходів наведена в табл. 16.2. Розглядати основні методи необхідно разом із параметрами, що впливають на ймовірність виявлення та продуктивність.

Таблиця 16.2 – Класифікація методів виявлення вибухонебезпечних предметів за фізичними принципами

| № з/п | Метод | Характеристика |
|---|---|---|
| 1 | Механічний | Контактний |
| | | Механізований |
| 2 | Електромагнітний | Радіохвильовий |
| | | Оптичний |
| | | Рентгенівський |
| | | Гамма-випромінювання |
| 3 | Хімічний | Газоаналітичний |
| | | Біофізичний |
| 4 | Магнітний | Магнітометричний |
| 5 | Акустичний | Сейсмоакустичний |

До *механічних методів* виявлення та знешкодження ВНП належать ручні, коли пошук і розмінування виконуються безпосередньо людиною, і механізовані, що виконуються спеціальними броньованими машинами. Сучасні машини розмінування є більш безпечними та ефективними в розмінуванні ВНП із вмістом вибухівки до 15 кг (наприклад, міни, саморобні вибухові пристрої та касетні боєприпаси). Принцип дії цього методу ґрунтується на механічному пошуку та знешкодженні ВНП. Робочим елементом машин для розмінування є приводи з ціпками, фрезами, культиваторами та спеціальними захватами [16]. Для підвищення якості знешкодження застосовуються комбіновані системи, наприклад, культиватора та ціпків. Такі платформи багатофункціональні, на них можна встановлювати різні інструменти, системи пошуку, навігації, дистанційного керування тощо. Основні технічні характеристики розглянемо на прикладі платформи MineWolf MW370 компанії Pearson Engineering: вага 23 т, ширина очищення 2,75 м, глибина очищення до 350 мм, швидкість очищення до 2,3 км/год, продуктивність розмінування до 30 000 м²/день, витрати палива 40–50 л/г і відстань дистанційного керування до 1000 м [17].

Отже, механічний метод виявлення та розмінування є простим, але забезпечує високу ймовірність виявлення та знешкодження ВНП, а потужний броньований захист і системи дистанційного керування зменшують ризик

травмування технічного персоналу.

Недоліки:

- незначна продуктивність та обмеженість використання залежно від рельєфу (неможливість працювати на мокрому та кам'янистому ґрунті, на уклонах понад 35°);
- високі вартість обладнання та витрати на виконання робіт;
- певні пошкодження екології.

*Електромагнітні методи* – це загальна назва групи методів, що працюють у різних частотних діапазонах і широко застосовуються для виявлення, побудови зображень та визначення властивостей об'єктів, що розташовані, зокрема, в оптично непрозорих середовищах, таких як ґрунт, бетон, цегляна кладка, асфальт, камінь, дерево та лід. Виявлення ВНП за допомогою електромагнітних методів базується на відмінності електромагнітних властивостей об'єкта та перешкод.

Умовно електромагнітне випромінювання, залежно від використовуваних частот, поділяється на: радіохвильове, оптичне, іонізуюче (рентгенівське та гамма-) випромінювання.

Системи, створені на базі цих методів, відрізняються робочою частотою, смугою електромагнітного спектра, типом сигналів, що передаються, інтерпретацією відбитих сигналів, типом передавача та приймача, а також алгоритмами оброблення.

*Радіохвильовий метод* – це найбільш поширений метод виявлення. На базі цього методу побудовані металодетектори (Metal Detector, MD), георадари (Ground Penetrating Radar, GPR), мікрохвильові радари (MWR), радари міліметрового діапазону (MMWR), радари електроімпедансної томографії (EIT).

*Метод електромагнітної індукції* (EMI) використовується в металодетекторах. Переваги цього методу – здатність виявляти металеві предмети розміром меншим за 1 см на глибині 50 см [18], не залежить від погодних умов і вологості ґрунту, має низьку вартість. Недоліки методу:

- нездатність виявляти ВНП із незначним вмістом металу (наприклад, у пластиковому корпусі);
- неможливість розрізнити ВНП та металеві уламки, що спричиняє високий відсоток помилкових тривог;
- мала дистанція пошуку.

Георадари, радіолокатори підповерхневого зондування або підповерхневі радіолокатори – це загальна назва радіолокаційних пристроїв, що реалізують технології використання електромагнітних хвиль для побудови зображень і визначення властивостей об'єктів, що розташовані в оптично непрозорих середовищах, таких як, наприклад, ґрунт, бетон, цегла, асфальт, камінь, дерево та лід. Зазвичай, радіолокатор такого типу з дальністю дії 1 м працює в діапазоні частот від 300 МГц до 3300 МГц [19].

Для підвищення ймовірності виявлення та мінімізації хибних спрацьовувань застосовується поєднання методів: георадара та

високочутливого металодетектора. Ці технології успішно використовуються для створення ручних мобільних приладів.

Розглянуті два методи реалізовані в міношукачі AN/PSS-14, спеціально розробленому для армії США. Додатково було застосовано алгоритми оброблення інформації, що забезпечило високі технічні характеристики за умови малої ваги: імовірність виявлення до 98,7 %, глибина виявлення ВНП до 300 мм, швидкість пошуку 3,2 м/хв, відстань сканування до 10 см, дистанційне керування не передбачено [20].

*Мікрохвильові радари* основані на використанні коротких радіоімпульсів та вимірюванні часу повернення відбитків. Відображення виникають на межах матеріалів із різною діелектричною проникністю. Підвищення частоти передачі забезпечує підвищення роздільної здатності, але водночас зростають і втрати в перешкодах.

*Оптичний метод*. Випромінювання в оптичному діапазоні (довжина хвилі 380–780 нм, частота 7,89·1014–3,84·1014Гц) умовно поділене на ультрафіолетове, видиме та інфрачервоне. Методи, що застосовують фізичні властивості цього випромінювання, успішно використовуються для виявлення і розпізнавання ВНП.

Ультрафіолетове випромінювання охоплює діапазон довжин хвиль 100–400 нм. У цьому діапазоні прямих демаскувальних ознак ВНП не виявлено, але в процесі застосування певного зовнішнього впливу, можуть з'являтися додаткові демаскувальні ознаки. Наприклад, у разі розпилення над забрудненою територією спеціального штаму бактерій, які проростають за декілька годин і флуоресціюють під ультрафіолетовим випромінюванням за наявності в ґрунті вибухових речовин [14].

Видиме випромінювання, що використовується для виявлення ВНП, передбачає захоплення світла у видимому діапазоні хвиль за допомогою оптичної системи для формування зображень. Використання сучасних широкоформатних багатоспектральних фотокамер дає змогу обстежувати значні території за короткий термін. Швидкість обстеження визначається швидкістю платформи, на якій розташовані оптичні сенсори. У разі використання літальних апаратів швидкість обстеження може перевищувати 100 км/год.

ВМС США продемонстрували прототип єдиної системи багатоцільового виявлення мін із повітря (SMAMD), розроблений компанією BAE Systems. У системі SMAMD використовується набір бортових оптичних датчиків, розміщених на борту безпілотного літального апарата (БПЛА) MQ_8C Fire Scout [21].

Обмеженням цього методу є те, що виявляти можна тільки ВНП, які розташовані на поверхні ґрунту. Також на якість виявлення впливають погодні умови та наявність маскувальних факторів (камуфляж, рослинність тощо).

Використання інфрачервоного випромінювання для виявлення ВНП ґрунтується на наявності різниці теплових характеристик між похованими об'єктами та навколишнім ґрунтом, що призводить до різниці температур між похованим об'єктом і ґрунтом. Цей температурний контраст вимірюється за

допомогою термографічної камери, яка виявляє випромінювання в інфрачервоному діапазоні електромагнітного спектра.

Перевагами методу є те, що він пасивний, отже, не впливає на системи керування ВНП, які можуть спричинити вибух; також цей метод дозволяє підвищувати швидкість / продуктивність обстеження за допомогою застосування БПЛА як платформи.

Недоліки: на якість виявлення впливають параметри навколишнього середовища (сонячне світло, дощ та ін.) та перешкоди (покривний шар ґрунту, рослинність тощо). Це значно звужує можливості застосування.

Рентгенівське випромінювання. Фотони характеристичного (тобто що випускається під час переходів в електронних оболонках атомів) рентгенівського випромінювання мають енергію від 10 еВ до 250 кеВ, що відповідає випромінюванню з частотою від 3*1016 до 3*1019 Гц і довжиною хвилі 0,005–100 нм. Цей метод ґрунтується на тому, що спостережувані частоти залежать від взаємодії між електричним квадрупольним моментом ядра та градієнтом електричного поля, що створюється в ядерному центрі зовнішніми зарядами. Усі звичайні вибухові речовини містять квадрупольне ядро, яке генерує три набори резонансних частот, забезпечуючи однозначний метод виявлення та ідентифікації вибухової речовини [22]. Він є похідним від ядерного магнітного резонансу і використовується без зовнішнього магнітного поля. Недоліком методу є виявлення тільки однієї групи, яка має бути відома заздалегідь.

Гамма-випромінювання. Ядерно-фізичні методи розрізняють за типом і енергією, що використовує джерело нейтронів, а також видом і енергією вторинного гамма-випромінювання, яке виникає під час взаємодії нейтрона з об'єктом пошуку – азотом (вуглецем чи киснем), що міститься у вибуховій речовині.

Вадою цих приладів, що випливає із фізичного принципу їхньої роботи, можна вважати те, що вони не мають вибірковості щодо вибухових або наркотичних речовин.

*Хімічний метод.* Широке застосування набув хімічний аналіз повітря на наявність випарів у місцях розташування ВНП. Він базується на виявленні та кількісній оцінці специфічних хімічних вибухових речовин та їхніх компонентів, які містяться у ВНП і дифундують у навколишнє середовище. Цей метод дає змогу виявляти сліди вибухових речовин у ґрунті або в повітрі в місцях знаходження ВНП. Дифузія речовин на поверхню відбувається в малих кількостях і залежить від конструкції ВНП, що вкривають середовище та погодні умови.

Ураховуючи малі концентрації речовин, неоднорідність поширення, різноманітні маскувальні фактори, необхідно застосовувати дуже чутливі датчики. Вважаємо, доцільною є думка, що цей метод містить газоаналітичні та біофізичні методи.

Газоаналітичні прилади є досить численним класом і забезпечують високу чутливість. Газоаналітичні методи здебільшого стосуються випарів тротилу, гексогену та ПЕТ; тому їх можна розглядати як підземні джерела пари.

Ця пара може транспортуватися за допомогою таких явищ, як молекулярна дифузія та процеси турбулентності. Ідея цього методу полягає у створенні газових датчиків, здатних виявляти малі концентрації хімічних речовин ВНП. Існують певні обмеження в цій галузі досліджень, що пояснюється неможливістю встановити мінімальний рівень виявлення через мінливу природу парів.

*Біофізичні методи*. Необхідно наголосити на успішному використанні тварин для пошуку ВНП за хімічними ознаками. Для виявлення ВНП ще за часів Другої світової війни застосовувалися собаки (Mine detection dog, MDD). Окрім собак, у різних країнах ведуться дослідження з використання інших тварин для пошуку ВНП, які більш пристосовані для певних територій і дозволяють зменшити витрати на вирощування та підготовку [12].

За програмою HeroRATs ведуться роботи з використання африканських гігантських мішкоподібних щурів (Mine detection rat, MDR), які допомагають знаходити наземні міни [23]. Навчені щури можуть обстежити територію розміром із тенісний корт (23,77 х 10,97 м) за 30 хв. Також вивчаються особливості використання для пошуку ВНП інших тварин та комах. Необхідно зазначити, що тварини мають і кращі газоаналітичні здібності, ніж електронні газоаналізатори. Це дає змогу виявляти вибухові речовини в нижчих концентраціях та з більшою ймовірністю.

За інформацією Marshall Legacy Institute (MLI) [24], MDD-команди зазвичай здійснюють пошук не тільки у 30 разів швидше, ніж команди, які використовують ручний спосіб пошуку, а й безпечніше. Жоден із фахівців з MDD не загинув під час операцій з розмінування.

Штучне виявлення випарів конкурує з тваринами або використовується разом із ними. Однак тварини більш чутливі та можуть виявляти багато різних запахів одночасно, що досить важко відтворити штучно.

Окремо необхідно наголосити на перспективах застосування тварин у створенні інтелектуальних автоматизованих систем виявлення і знешкодження ВНП. Концепція основана на розміщені на тваринах мобільної інтелектуальної системи, яка забезпечує навігацію, дистанційне наведення та спостереження. Такі системи підвищують імовірності виявлення ВНП.

Недоліками газоаналітичних методів є:

- необхідність переналаштування / перенавчання в разі появи нових хімічних компонентів у ВНП;
- велика залежність якості вимірювань від навколишнього середовища та погодних умов;
- робота в небезпечній зоні (безпосередньо біля ВНП).

*Магнітометричний метод*. Використання магнітометрів як металошукачів основане на явищі локального спотворення природного магнітного поля Землі феромагнітними матеріалами, наприклад залізом. Порівняно з розглянутими вище принципами, магнітометри мають набагато більшу дальність виявлення залізних предметів. Залізні предмети створюють аномалії, які фіксуються цими приладами.

Цей принцип розвивається з метою розроблення якісного відеоаналізу, схованого під поверхнею об'єкта, силові магнітні лінії від якого знімають магнітометром.

Основні переваги цього методу:

- можливість виявлення у природних покривних середовищах;
- більша глибина й висока швидкість пошуку.

Недоліки методу:

- неможливість детектування діелектричних матеріалів;
- низька вибірковість та завадостійкість.

*Акустичний метод*. Акустичні хвилі можуть бути ефективним інструментом для виявлення та ідентифікації наземних мін. Ультразвукові методи та методи перетворення акустичних даних на сейсмічні (A/S) є поширеними методами акустичного виявлення. Недоліками цього методу є низька роздільна здатність та залежність від щільності ґрунту. Тому цей метод має малу ефективність для пошуку ВНП, особливо за наявності декількох перешкод із різними властивостями, наприклад, повітря-ґрунт.

За результатами аналізу методів виявлення та розпізнавання, визначені основні характеристики та виконане порівняння. Результати порівняння методів виявлення за основними характеристиками наведені в табл. 16.3.

Таблиця 16.3 – Порівняння методів виявлення вибухонебезпечних предметів

| Характеристика | Контактний | Механізований | Радіохвильовий | Оптичний | Рентгенівське | Гамма | Газоаналітичний | Біофізичний | Магнітометричний | Сейсмоакустичні |
|---|---|---|---|---|---|---|---|---|---|---|
| Тип взаємодії з ВНП (А – активний, П – пасивний) | А | А | А | П | А | А | П | П | П | А |
| Тип платформи (Ст – стаціонарний, Моб – мобільний) | Ст | Моб | Моб | Моб | Моб | Моб | Моб | Моб | Моб | Ст |
| Потенційна продуктивність (Med – Medium) | Low | Med | High | High | Med | Med | Med | Med | High | Low |
| Підтримка інформаційних технологій | | | | | | | | | | |
| Оброблення даних та їхнє зберігання | - | - | + | + | + | + | - | - | + | + |
| Доступ до мережі та покриття | - | - | + | + | + | + | - | - | + | - |
| Передача даних у режимі реального часу | - | - | + | + | + | + | - | - | + | - |
| Параметри якості | | | | | | | | | | |
| Безпека під час виконання робіт | Low | Med | High | Low | High | High | High | High | Low | Med |
| Імовірність виявлення | High | High | Med | Low | High | High | High | High | Med | Med |
| Роздільна здатність (вибірковість) | High | Low | Med | Med | High | High | High | Low | Low | Low |
| Надійність | High | High | High | High | High | High | Low | Low | Med | Med |
| Економічні показники | | | | | | | | | | |
| Вартість | Low | Med | Med | Med | High | High | Med | Low | Med | Med |

16.4. Балансування навантаження між безпілотними літальними апаратами флоту під час виконання ним завдань з виявлення вибухонебезпечних предметів при використанні автоматичних енерговідновлювальних станцій

Залучення флоту БПЛА для виявлення ВНП може включати як завдання з виявлення ВНП, так і утворення бездротової літаючої мережі (БПЛА-БМ) для передавання інформації про ВНП керівництву групи, відповідальної за їх виявлення

Під час функціонування БПЛА-БМ, БПЛА, що входять до її складу, можуть повертатися після заміни (заряджання) батарей на автоматичній енерговідновлювальній станцій (АЕВС) до попередньої точки свого розташування у БПЛА-БМ (здійснювати польоти за одним і тим же маршрутом), або маршрути кожного разу можуть обиратись за визначеним правилом.

Розглянемо функціонування БПЛА-БМ, що складається з п'яти БПЛА мультироторного типу (М-БПЛА) та використовує одну АЕВС [25].

Приймемо наступні припущення.

1. П'ять М-БПЛА періодично відправляються своєю зміною до АЕВС для заміни батарей.

2. Відстань, яку долає М-БПЛА$_i$ ($i = \overline{1,5}$) за *z*-й цикл чергування ($z = \overline{1,\phi}$), рухаючись за маршрутом «АЕВС (точка 0) □ точка k □ АЕВС (точка 0)» розраховується наступним чином:

$$S_z^{\text{М-БПЛА}_i} = S_{0,k,0} = 2S_{0,k}\,(\text{км}), \tag{16.1}$$

де $S_{0,k}$ – відстань від точки 0 до точки k;

k – точка розташування М-БПЛА$_i$ у складі БПЛА-БМ під час *z*-го циклу чергування.

3. Сумарна відстань, яку М-БПЛА$_i$ долає з 1-го по *z*-й цикл чергування визначається у такий спосіб:

$$S_{(\overline{1,z})}^{\text{М-БПЛА}_i} = \sum_{\varphi(z)=1}^{z} S_{\varphi(z)}^{\text{М-БПЛА}_i}. \tag{16.2}$$

4. Точка розташування М-БПЛА$_i$ у складі БПЛА-БМ під час кожного наступного циклу чергування може змінюватися (рис. 16.3). Наприклад, М-БПЛА$_1$, який рухався за маршрутом «точка 0 □ точка 1 □ точка 0» у першому циклі чергування, у другому циклі рухається за маршрутом «точка 0 □ точка 4 □ точка 0».

5. У якості правила при плануванні маршрутів польотів М-БПЛА при функціонуванні БПЛА-БМ пропонується використовувати критерій максиміну

– вибір точок розташування М-БПЛА у складі БПЛА-БМ має здійснюватися з дотриманням умови

$$\max\left(\sum_{z=1}^{\phi} S_z^{\text{М-БПЛА}_\alpha} - \sum_{z=1}^{\phi} S_z^{\text{М-БПЛА}_\beta}\right) \to \min; \forall \alpha,\beta\ (\alpha \neq \beta);\ \alpha = \overline{1,n};\ \beta = \overline{1,n}\,. \qquad (16.3)$$

Ця задача може бути зведена до задачі про пошук найкоротшого шляху для кожного окремого М-БПЛА із забезпеченням балансування навантаження між М-БПЛА в ході виконання поставлених завдань.

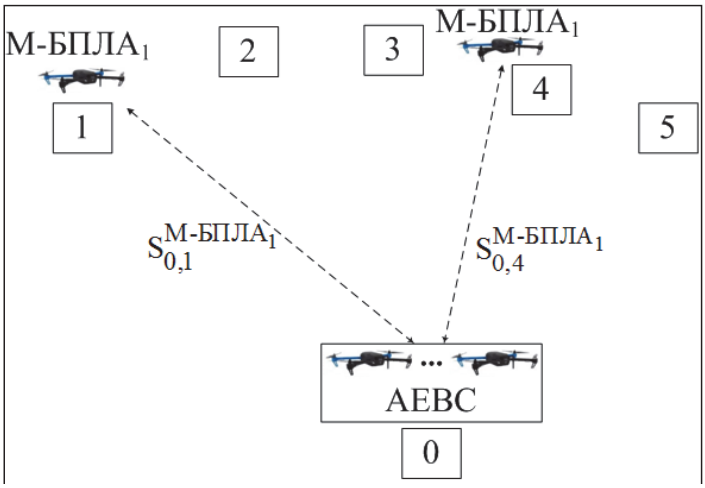



Рисунок 16. 3 – Приклад зміни точки розташування М-БПЛА у складі БПЛА-БМ

Існує велика кількість методів пошуку найкоротшого шляху [26–31], які базуються на аналізі графів. Суттєвим недоліком цих методів є пошук найкоротшого шляху без урахування циклічних повторювань маршрутів. Тому для вирішення цієї задачі пропонується застосовувати наступний метод. Припустимо, що під час першого циклу чергування кожен із п’яти М-БПЛА, що входять до складу БПЛА-БМ, рухається з АЕВС до визначеного довільним чином місця свого розташування у складі БПЛА-БМ та повертається на АЕВС. Починаючи з другого циклу здійснюється балансування навантаження між М-БПЛА, яке полягає в корегуванні маршруту.

Розглянемо процедуру корегування маршрутів під час кожного наступного циклу чергування М-БПЛА.

Задамо наступні вихідні дані.

1. Відстані від можливих точок розташування М-БПЛА у складі БПЛА-БМ до АЕВС дорівнюють: $S_{0,1}$ = 0,54 км; $S_{0,2}$= 0,51 км; $S_{0,3}$= 0,50 км; $S_{0,4}$=

0,51 км; $S_{0,5} = 0{,}54$ км.

2. Кількість циклів чергування z = 4.

Приймемо припущення про те, що в першому циклі чергування номера точок відповідають номерам М-БПЛА, які їх відвідують. Ці маршрути показані у таблиці 16.4.

Таблиця 16.4 – Маршрути польоту для кожного М-БПЛА під час першого циклу чергування

| М-БПЛА | Маршрут польоту |
|---|---|
| М-БПЛА$_1$ | «точка 0 □ точка 1 □ точка 0» |
| М-БПЛА$_2$ | «точка 0 □ точка 2 □ точка 0» |
| М-БПЛА$_3$ | «точка 0 □ точка 3 □ точка 0» |
| М-БПЛА$_4$ | «точка 0 □ точка 4 □ точка 0» |
| М-БПЛА$_5$ | «точка 0 □ точка 5 □ точка 0» |

Визначимо маршрут польоту та його протяжність для кожного М-БПЛА під час другого циклу чергування.

1. Побудуємо одномірні масиви:

– відстаней, пройдених кожним М-БПЛА за 1-й цикл чергування:

$$S_{(1)}^{\text{М-БПЛА}} = \begin{pmatrix} S_{(1)}^{\text{М-БПЛА}_1} \\ S_{(1)}^{\text{М-БПЛА}_2} \\ S_{(1)}^{\text{М-БПЛА}_3} \\ S_{(1)}^{\text{М-БПЛА}_4} \\ S_{(1)}^{\text{М-БПЛА}_5} \end{pmatrix} = \begin{pmatrix} S_{0,1,0} \\ S_{0,2,0} \\ S_{0,3,0} \\ S_{0,4,0} \\ S_{0,5,0} \end{pmatrix} = \begin{pmatrix} 2 \cdot S_{0,1} \\ 2 \cdot S_{0,2} \\ 2 \cdot S_{0,3} \\ 2 \cdot S_{0,4} \\ 2 \cdot S_{0,5} \end{pmatrix} = \begin{pmatrix} 2 \cdot 0{,}54 \\ 2 \cdot 0{,}51 \\ 2 \cdot 0{,}50 \\ 2 \cdot 0{,}51 \\ 2 \cdot 0{,}54 \end{pmatrix} = \begin{pmatrix} 1{,}08 \\ 1{,}02 \\ 1{,}00 \\ 1{,}02 \\ 1{,}08 \end{pmatrix};$$

– довжин маршрутів «точка 0 □ точка $k$ □ точка 0», $k = \overline{1,5}$:

$$S^{\text{маршр}} = \begin{pmatrix} S_{0,1,0} \\ S_{0,2,0} \\ S_{0,3,0} \\ S_{0,4,0} \\ S_{0,5,0} \end{pmatrix} = \begin{pmatrix} 2 \cdot S_{0,1} \\ 2 \cdot S_{0,2} \\ 2 \cdot S_{0,3} \\ 2 \cdot S_{0,4} \\ 2 \cdot S_{0,5} \end{pmatrix} = \begin{pmatrix} 2 \cdot 0{,}54 \\ 2 \cdot 0{,}51 \\ 2 \cdot 0{,}50 \\ 2 \cdot 0{,}51 \\ 2 \cdot 0{,}54 \end{pmatrix} = \begin{pmatrix} 1{,}08 \\ 1{,}02 \\ 1{,}00 \\ 1{,}02 \\ 1{,}08 \end{pmatrix}.$$

2. Зробимо упорядкування:

– масиву $S_{(1)}^{\text{М-БПЛА}}$ шляхом розташування його елементів у порядку зменшення їхніх значень із отриманням нового одномірного масиву:

$$S_{(1)}^{\text{М-БПЛА}}(\downarrow)=\begin{pmatrix} S_{(1)}^{\text{М-БПЛА}_1} \\ S_{(1)}^{\text{М-БПЛА}_5} \\ S_{(1)}^{\text{М-БПЛА}_2} \\ S_{(1)}^{\text{М-БПЛА}_4} \\ S_{(1)}^{\text{М-БПЛА}_3} \end{pmatrix}=\begin{pmatrix} 1,08 \\ 1,08 \\ 1,02 \\ 1,02 \\ 1,00 \end{pmatrix};$$

– масиву $S^{\text{маршр}}$ шляхом розташування його елементів у порядку збільшення їхніх значень із отриманням нового одномірного масиву:

$$S^{\text{маршр}}(\uparrow)=\begin{pmatrix} S_{0,3,0} \\ S_{0,2,0} \\ S_{0,4,0} \\ S_{0,1,0} \\ S_{0,5,0} \end{pmatrix}=\begin{pmatrix} 1,00 \\ 1,02 \\ 1,02 \\ 1,08 \\ 1,08 \end{pmatrix}.$$

Відзначимо, що упорядкування масиву $S^{\text{маршр}}$ з метою отримання нового масиву $S^{\text{маршр}}(\uparrow)$ здійснюється тільки на цьому етапі розрахунків. У подальшому всюди використовується масив $S^{\text{маршр}}(\uparrow)$.

3. Замінюючи в елементах масиву $S_{(1)}^{\text{М-БПЛА}}(\downarrow)$ нижній індекс з (1) на (2) та ставлячи цьому масиву у відповідність масив $S^{\text{маршр}}(\uparrow)$, отримуємо новий масив, що містить інформацію про маршрути, за якими повинен рухатися кожен М-БПЛА під час другого циклу чергування та їх протяжність:

$$S_{(2)}^{\text{М-БПЛА}}=\begin{pmatrix} S_{(2)}^{\text{М-БПЛА}_1} \\ S_{(2)}^{\text{М-БПЛА}_5} \\ S_{(2)}^{\text{М-БПЛА}_2} \\ S_{(2)}^{\text{М-БПЛА}_4} \\ S_{(2)}^{\text{М-БПЛА}_3} \end{pmatrix}=\begin{pmatrix} S_{0,3,0} \\ S_{0,2,0} \\ S_{0,4,0} \\ S_{0,1,0} \\ S_{0,5,0} \end{pmatrix}=\begin{pmatrix} 1,00 \\ 1,02 \\ 1,02 \\ 1,08 \\ 1,08 \end{pmatrix}$$

або в порядку збільшення номерів М-БПЛА:

$$S_{(2)}^{\text{М-БПЛА}} = \begin{pmatrix} S_{(2)}^{\text{М-БПЛА}_1} \\ S_{(2)}^{\text{М-БПЛА}_2} \\ S_{(2)}^{\text{М-БПЛА}_3} \\ S_{(2)}^{\text{М-БПЛА}_4} \\ S_{(2)}^{\text{М-БПЛА}5} \end{pmatrix} = \begin{pmatrix} S_{0,3,0} \\ S_{0,4,0} \\ S_{0,5,0} \\ S_{0,1,0} \\ S_{0,2,0} \end{pmatrix} = \begin{pmatrix} 1,00 \\ 1,02 \\ 1,08 \\ 1,08 \\ 1,02 \end{pmatrix}.$$

Для наочності представимо маршрути польоту кожного М-БПЛА під час другого циклу чергування у вигляді таблиці 16.5.

Таблиця 16.5 – Маршрути польоту кожного М-БПЛА під час другого циклу чергування

| М-БПЛА | Маршрут польоту |
|---|---|
| М-БПЛА$_1$ | «точка 0 □ точка 3 □ точка 0» |
| М-БПЛА$_2$ | «точка 0 □ точка 4 □ точка 0» |
| М-БПЛА$_3$ | «точка 0 □ точка 5 □ точка 0» |
| М-БПЛА$_4$ | «точка 0 □ точка 1 □ точка 0» |
| М-БПЛА$_5$ | «точка 0 □ точка 2 □ точка 0» |

Визначимо маршрут польоту та його протяжність для кожного М-БПЛА під час третього циклу чергування.

1. Побудуємо одномірний масив відстаней, пройдених кожним М-БПЛА за перший та другий цикли чергування

$$S_{(\overline{1,2})}^{\text{М-БПЛА}} = \begin{pmatrix} S_{(\overline{1,2})}^{\text{М-БПЛА}_1} \\ S_{(\overline{1,2})}^{\text{М-БПЛА}_2} \\ S_{(\overline{1,2})}^{\text{М-БПЛА}_3} \\ S_{(\overline{1,2})}^{\text{М-БПЛА}_4} \\ S_{(\overline{1,2})}^{\text{М-БПЛА}_5} \end{pmatrix} = \begin{pmatrix} 1,08 \\ 1,02 \\ 1,00 \\ 1,02 \\ 1,08 \end{pmatrix} + \begin{pmatrix} 1,00 \\ 1,02 \\ 1,08 \\ 1,08 \\ 1,02 \end{pmatrix} = \begin{pmatrix} 2,08 \\ 2,04 \\ 2,08 \\ 2,10 \\ 2,10 \end{pmatrix}.$$

2. Зробимо упорядкування масиву $S_{(\overline{1,2})}^{\text{М-БПЛА}}$ шляхом розташування його елементів в порядку зменшення їхніх значень із отриманням нового одномірного масиву:

$$S^{\text{М-БПЛА}}_{(\overline{1,2})}(\downarrow)=\begin{pmatrix} S^{\text{М-БПЛА}_4}_{(\overline{1,2})} \\ S^{\text{М-БПЛА}_5}_{(\overline{1,2})} \\ S^{\text{М-БПЛА}_1}_{(\overline{1,2})} \\ S^{\text{М-БПЛА}_3}_{(\overline{1,2})} \\ S^{\text{М-БПЛА}_2}_{(\overline{1,2})} \end{pmatrix}=\begin{pmatrix} 2,10 \\ 2,10 \\ 2,08 \\ 2,08 \\ 2,04 \end{pmatrix}.$$

3. Замінюючи в елементах масиву $S^{\text{М-БПЛА}}_{(\overline{1,2})}(\downarrow)$ нижній індекс з $(\overline{1,2})$ на (3) та ставлячи цьому масиву у відповідність масив $S^{\text{маршр}}(\uparrow)$, отримуємо новий масив, що містить інформацію про маршрути, за якими повинен рухатися кожен М-БПЛА під час третього циклу чергування та їх протяжність:

$$S^{\text{М-БПЛА}}_{(3)}=\begin{pmatrix} S^{\text{М-БПЛА}_4}_{(3)} \\ S^{\text{М-БПЛА}_5}_{(3)} \\ S^{\text{М-БПЛА}_1}_{(3)} \\ S^{\text{М-БПЛА}_3}_{(3)} \\ S^{\text{М-БПЛА}_2}_{(3)} \end{pmatrix}=\begin{pmatrix} S_{0,3,0} \\ S_{0,2,0} \\ S_{0,4,0} \\ S_{0,1,0} \\ S_{0,5,0} \end{pmatrix}=\begin{pmatrix} 1,00 \\ 1,02 \\ 1,02 \\ 1,08 \\ 1,08 \end{pmatrix}$$

або в порядку збільшення номерів М-БПЛА:

$$S^{\text{М-БПЛА}}_{(3)}=\begin{pmatrix} S^{\text{М-БПЛА}_1}_{(3)} \\ S^{\text{М-БПЛА}_2}_{(3)} \\ S^{\text{М-БПЛА}_3}_{(3)} \\ S^{\text{М-БПЛА}_4}_{(3)} \\ S^{\text{М-БПЛА5}}_{(3)} \end{pmatrix}=\begin{pmatrix} S_{0,4,0} \\ S_{0,5,0} \\ S_{0,1,0} \\ S_{0,3,0} \\ S_{0,2,0} \end{pmatrix}=\begin{pmatrix} 1,02 \\ 1,08 \\ 1,08 \\ 1,00 \\ 1,02 \end{pmatrix}.$$

Для наочності представимо маршрути польоту кожного М-БПЛА під час другого циклу чергування у вигляді таблиці 16.6.

Таблиця 16.6 – Маршрути польоту кожного М-БПЛА під час третього циклу чергування

| М-БПЛА | Маршрут польоту |
|---|---|
| М-БПЛА$_1$ | «точка 0 □ точка 4 □ точка 0» |
| М-БПЛА$_2$ | «точка 0 □ точка 5 □ точка 0» |
| М-БПЛА$_3$ | «точка 0 □ точка 1 □ точка 0» |
| М-БПЛА$_4$ | «точка 0 □ точка 3 □ точка 0» |
| М-БПЛА$_5$ | «точка 0 □ точка 2 □ точка 0» |

Визначимо маршрут польоту та його протяжність для кожного М-БПЛА під час четвертого циклу чергування.

1. Побудуємо одномірний масив відстаней, пройдених кожним М-БПЛА за перший-третій цикли чергування:

$$S_{(\overline{1,3})}^{\text{М-БПЛА}} = \begin{pmatrix} 2,08 \\ 2,04 \\ 2,08 \\ 2,10 \\ 2,10 \end{pmatrix} + \begin{pmatrix} 1,02 \\ 1,08 \\ 1,08 \\ 1,00 \\ 1,02 \end{pmatrix} = \begin{pmatrix} 3,10 \\ 3,12 \\ 3,16 \\ 3,10 \\ 3,12 \end{pmatrix}.$$

2. Зробимо упорядкування масиву $S_{(\overline{1,3})}^{\text{М-БПЛА}}$ шляхом розташування його елементів в порядку зменшення їхніх значень із отриманням нового одномірного масиву:

$$S_{(\overline{1,3})}^{\text{М-БПЛА}}(\downarrow) = \begin{pmatrix} S_{(\overline{1,3})}^{\text{М-БПЛА}_3} \\ S_{(\overline{1,3})}^{\text{М-БПЛА}_2} \\ S_{(\overline{1,3})}^{\text{М-БПЛА}_5} \\ S_{(\overline{1,3})}^{\text{М-БПЛА}_1} \\ S_{(\overline{1,3})}^{\text{М-БПЛА}_4} \end{pmatrix} = \begin{pmatrix} 3,16 \\ 3,12 \\ 3,12 \\ 3,10 \\ 3,10 \end{pmatrix}.$$

3. Замінюючи в елементах масиву $S_{(\overline{1,3})}^{\text{М-БПЛА}}(\downarrow)$ нижній індекс з $(\overline{1,3})$ на (4) та ставлячи цьому масиву у відповідність масив $S^{\text{маршр}}(\uparrow)$, отримуємо новий масив, що містить інформацію про маршрути, за якими повинен рухатися кожен М-БПЛА під час третього циклу чергування та їх протяжність:

$$S_{(4)}^{\text{М-БПЛА}} = \begin{pmatrix} S_{(4)}^{\text{М-БПЛА}_3} \\ S_{(4)}^{\text{М-БПЛА}_2} \\ S_{(4)}^{\text{М-БПЛА}_5} \\ S_{(4)}^{\text{М-БПЛА}_1} \\ S_{(4)}^{\text{М-БПЛА}_4} \end{pmatrix} = \begin{pmatrix} S_{0,3,0} \\ S_{0,2,0} \\ S_{0,4,0} \\ S_{0,1,0} \\ S_{0,5,0} \end{pmatrix} = \begin{pmatrix} 1,00 \\ 1,02 \\ 1,02 \\ 1,08 \\ 1,08 \end{pmatrix}.$$

або в порядку збільшення номерів М-БПЛА:

$$S_{(4)}^{\text{М-БПЛА}} = \begin{pmatrix} S_{(4)}^{\text{М-БПЛА}_1} \\ S_{(4)}^{\text{М-БПЛА}_2} \\ S_{(4)}^{\text{М-БПЛА}_3} \\ S_{(4)}^{\text{М-БПЛА}_4} \\ S_{(4)}^{\text{М-БПЛА5}} \end{pmatrix} = \begin{pmatrix} S_{0,1,0} \\ S_{0,2,0} \\ S_{0,3,0} \\ S_{0,5,0} \\ S_{0,4,0} \end{pmatrix} = \begin{pmatrix} 1,08 \\ 1,02 \\ 1,00 \\ 1,08 \\ 1,02 \end{pmatrix}.$$

Для наочності представимо маршрути польоту кожного М-БПЛА під час четвертого циклу чергування у вигляді таблиці 16.7.

Таблиця 16.7 – Маршрути польоту кожного М-БПЛА під час четвертого циклу чергування

| М-БПЛА | Маршрут польоту |
|---|---|
| $\text{М-БПЛА}_1$ | «точка 0 □ точка 1 □ точка 0» |
| $\text{М-БПЛА}_2$ | «точка 0 □ точка 2 □ точка 0» |
| $\text{М-БПЛА}_3$ | «точка 0 □ точка 3 □ точка 0» |
| $\text{М-БПЛА}_4$ | «точка 0 □ точка 5 □ точка 0» |
| $\text{М-БПЛА}_5$ | «точка 0 □ точка 4 □ точка 0» |

Таким чином, продемонстровано можливість застосування запропонованого методу для балансування навантаження між п’ятьма М-БПЛА флоту, що несуть чергування у складі БПЛА-БМ, протягом чотирьох циклів їхнього чергування. Забезпечення такого балансування надасть можливість підвищити надійність застосування БПЛА флоту під час здійснення заходів щодо виявлення ВНП.

## 16.5. Концепція гарантованого виявлення і розпізнавання вибухонебезпечних предметів з використанням безпілотних літальних апаратів

Аналіз методів дає змогу зробити висновок про суперечність між (а) необхідністю підвищення продуктивності обстеження територій із високою ймовірністю наявності різних типів вибухонебезпечних предметів та забезпечення необхідної достовірності їхньої ідентифікації для знешкодження, а також (б) наявністю методів і технологічних рішень мобільних роботизованих комплексів, а саме флотів БПЛА, з одного боку, та відсутністю цілісної концепції, методів, засобів і технологій створення і використання багатоцільових надійних і безпечних інтелектуальних систем БПЛА для пошуку та розпізнавання ВНП, з іншого боку [32].

Для подолання цієї суперечності важливо, насамперед, створити концепцію, на основі якої розробити методологію, моделі та інформаційні технології гарантованого виявлення і розпізнавання ВНП з використанням мобільних розподілених багатоверсійних інтелектуальних систем. Принципи побудови та застосування, математичні моделі та методи забезпечення надійності та безпечності таких систем запропоновано та досліджено в роботах [33, 34].

Концепція ґрунтується на двох положеннях, що поєднують прагнення забезпечити високу продуктивність і достовірність пошуку, виявлення та розпізнавання ВНП для їхнього подальшого знешкодження.

По-перше, високопродуктивне виявлення ВНП на визначеній території забезпечується шляхом:

- застосування багатофункційного флоту БПЛА, оснащеного різними типами інформаційно-вимірювальних засобів;
- планування оптимальних маршрутів руху БПЛА з урахуванням багатопараметричного покриття під час виконання поставлених завдань.

По-друге, висока достовірність виявлення і розпізнавання ВНП досягається шляхом:

- дворівневого інтелектуального оброблення інформації в розподіленій архітектурі літальних граничних обчислень (Flying Edge Computing) з урахуванням кореляції інформації, отриманої від БПЛА з різними інформаційно-вимірювальними засобами;
- донавчання відповідних нейромережних структур упродовж виконання завдань на різних територіях.

Запропонована концепція гарантованого пошуку та розпізнавання ВНП дає змогу сформулювати завдання, що необхідно розв'язати для її реалізації:

- розробити методологію (концепцію, принципи та структуру взаємозв'язків моделей, методів, програмно-апаратних засобів та інформаційних технологій) створення та використання надійних і безпечних ББІС пошуку та знешкодження ВНП;

 розробити системні моделі та комплекс показників ефективності надійних і безпечних ББІС;

 розробити та дослідити моделі та ієрархічні структури бортових комплексів ББІС для пошуку, ідентифікації та знешкодження ВНП із різними варіантами розподілу функцій та конфігурування апаратних і програмних компонентів;

 удосконалити методи пошуку та виявлення ВНП із застосуванням ББІС;

 розробити та вдосконалити методи планування використання ББІС пошуку та знешкодження ВНП на територіях із фіксованою конфігурацією та типами ВНП;

 розробити та дослідити моделі та методи підвищення надійності ББІС та їхніх компонентів;

 розробити структуру та засоби ІТ для створення, планування використання та забезпечення надійного функціонування ББІС пошуку та знешкодження ВНП;

 розробити та дослідити модель і метод аналізу та оцінювання ризику виникнення небезпечного стану «підрив на вибухонебезпечному предметі».

## 16.6. Висновки

За результатами виконаного аналізу методів виявлення ВНП можна зробити висновки, що:

 зростання кількості та інтенсивності озброєних конфліктів і війн у світі, висока інтенсивність застосування боєприпасів різних типів та інтенсивність використання призводить до збільшення площі територій, забруднених ВНП;

 низька продуктивність наявних методів не дозволяє швидко й ефективно розчищати забруднені ВНП території, що призводить до значної кількості уражень і загибелі людей унаслідок підриву ВНП;

 самостійне використання окремих методів виявлення не може суттєво підвищити ймовірність виявлення ВНП;

 для підвищення продуктивності та безпеки виконання робіт із пошуку та знешкодження ВНП доцільно використовувати безпілотні інтелектуальні платформи направлення інформаційно-вимірювальних засобів.

*Подальші дослідження* доцільно проводити таким чином: проаналізувати стан і тенденції розвитку методів, технологій і математичного апарата для створення та застосування систем виявлення й розпізнавання ВНП на базі флотів БПЛА та інших роботизованих засобів (ББІС); удосконалити методи пошуку та виявлення ВНП із застосуванням БПЛА; дослідити методи підвищення ефективності пошуку ВНП за допомогою використання програмно-апаратних засобів та інформаційних технологій.

Література

1. Landmine Monitor 2022. URL: http://www.the-monitor.org/en-gb/reports/2022/landmine-monitor-2022.aspx (дата звернення: 18.11.2022).

2. Ukraine: Mine Action – 5W Situation Report (as of 01 June 2022). URL: https://reliefweb.int/report/ukraine/ukraine-mine-action-5w-situation-report-01-june-2022 (дата звернення: 18.11.2022).

3. Kenny P. Landmines killed 7,073 in 2020, says UN institute. URL: https://www.aa.com.tr/en/world/landmines-killed-7-073-in-2020-says-un-institute/2417253 (дата звернення: 18.11.2022).

4. Аналіз виконання робіт щодо очищення території України від вибухонебезпечних предметів у 2021 році. URL: https://dsns.gov.ua/uk/protiminna-diyalnist/gumanitarne-rozminuvannya (дата звернення: 18.11.2022).

5. Огляд збитків від війни в сільському господарстві України: непряма оцінка пошкоджень. URL: https://kse.ua/wp-content/uploads/2022/06/Damages_report_issue1_ua-1.pdf (дата звернення: 18.11.2022).

6. Robledo L., Carrasco M., Mery D. A survey of land mine detection technology. *International Journal of Remote Sensing*. 2009. Vol. 30. Issue 9. P. 2399–2410. DOI: https://doi.org/10.1080/01431160802549435

7. Kasban H., Zahran O., Elaraby S. M., El-Kordy M., Abd El-Samie F. E. A comparative study of landmine detection techniques. *Sensing and Imaging*. 2010. Vol. 11. Issue 3. P. 89–112. DOI: https://doi.org/10.1007/s11220-010-0054-x

8. Гайдарли Г. С. Розмінування території і об'єктів інженерними підрозділами збройних сил України у міжнародних операціях з підтримання миру і безпеки (1992–2018): дис. канд. іст. наук: 20.02.22. Київ, 2020. 274 с.

9. Молочко С. М., Башинський В. Г., Каламурза О. Г., Журахов В. А. Аналіз сучасного стану, характеристик та перспектив розвитку датчиків виявлення вибухонебезпечних предметів, встановлених на БпАК. *Збірник наукових праць Державного науково-дослідного інституту випробувань і сертифікації озброєння та військової техніки*. 2021. № 2 (8). С. 80–90. DOI: https://doi.org/10.37701/dndivsovt.8.2021.09

10. Fernandez M. G., Lopez Y. A., Arboleya A. A., Valdes B. G., Vaqueiro Y. R., Andres F. L. H., Garcia A. P. Synthetic aperture radar imaging system for landmine detection using a ground penetrating radar on board a unmanned aerial vehicle. *IEEE Access*. 2018. Vol. 6. P. 45100–45112. DOI: https://doi.org/10.1109/ACCESS.2018.2863572

11. Pajares G. Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs). *Photogrammetric Engineering and Remote Sensing*. 2015. Vol. 81, Issue 4. P. 281–329. DOI: https://doi.org/10.14358/PERS.81.4.281

12. Field trials of the smart system and technical survey dogs in Cambodia: Final report 2021. URL: https://www.gichd.org/fileadmin/GICHD-resources/rec-documents/SMART_Cambodia_v13__1__01.pdf (дата звернення: 18.11.2022).

13. Filipi J., Stojnić V., Muštra M., Gillanders R. N., Jovanović V., Gajić S., Turnbull G. A., Babić Z., Kezić N., Risojević V. Honeybee-based biohybrid system for landmine detection. *Science of The Total Environment*. 2022. Vol. 803. DOI: https://doi.org/10.1016/j.scitotenv.2021.150041

14. Shemer B., Palevsky N., Yagur-Kroll S., Belkin S. Genetically engineered microorganisms for the detection of explosives' residues. *Frontiers in Microbiology*. 2015. Vol. 6. DOI: https://doi.org/10.3389/fmicb.2015.01175

15. Challenges for Mine Action due to russian aggression against Ukraine. URL: https://www.mineaction.org/sites/default/files/2.1.1_challenges_for_mine_action_in_ukraine.pdf (дата звернення: 18.11.2022).

16. A Study of Mechanical Application in Demining. URL: https://www.gichd.org/fileadmin/GICHD-resources/rec-documents/Mechanical_study_complete.pdf (дата звернення: 18.11.2022).

17. The MW370 is a powerful mine and route clearance platform used for the effective clearance of landmines across large areas. URL: https://www.pearson-eng.com/product/mw370/ (дата звернення: 18.11.2022).

18. van Verre W., Podd F. J., Daniels D. J., Peyton A. J. A Review of Passive and Active Ultra-Wideband Baluns for Use in Ground Penetrating Radar. *Remote Sensing*. 2021. Vol. 13. Issue 10. DOI: https://doi.org/10.3390/rs13101899

19. Song X., Liu T., Xiang D., Su Y. GPR Antipersonnel Mine Detection Based on Tensor Robust Principal Analysis. *Remote Sensing*. 2019. Vol. 11. Issue 8. DOI: https://doi.org/10.3390/rs11080984

20. Schweitzer K. M., Davis B. M., Pettijohn B. A., Clark R. D., Davison A. D., Staszewski J. J. Optimization of Army-Navy/Portable Special Search (AN/PSS)-14 Operator Training. URL: https://apps.dtic.mil/sti/pdfs/ADA457012.pdf (дата звернення: 18.11.2022).

21. U.S. Navy Demos MCM Equipment Prototype On MQ-8C. URL: https://www.navalnews.com/naval-news/2022/07/u-s-navy-demos-mcm-equipment-prototype-on-mq-8c (дата звернення: 18.11.2022).

22. Вижва С. А., Онищук І. І., Черняєв О. П. Ядерна геофізика: підручник. Київ. Видавничо-поліграфічний центр "Київський університет", 2012. 608 с.

23. Efficiency and Effectiveness Study using MDR capability. URL: https://www.gichd.org/fileadmin/GICHD-resources/rec-documents/APOPO-GICHD-Mine-Detection-Rats-30Jun2016.pdf (дата звернення: 18.11.2022).

24. Mine detection dog programs. URL: https://www.marshall-legacy.org/mine-detection-dog-programs (дата звернення: 18.11.2022).

25. Клюшніков І. М., Фесенко Г. В. Балансування навантаження між безпілотними літальними апаратами літаючої бездротової мережі у разі використання автоматичних обмінно-зарядних станцій. *Комунальне господарство міст*. 2020. № 1 (154). С. 113–119. DOI: 10.33042/2522-1809-2020-1-154-113-119.

26. Bauer R., Delling D., Sanders P., Schieferdecker D., Schultes D., Wagner D. Combining hierarchical and goal-directed speed-up techniques for Dijkstra's algorithm. *ACM Journal of Experimental Algorithmics*. 2010. Vol. 15, no. 2. P. 1–31. DOI: 10.1145/1671973.1671976.

27. Wang, Z., Han W., Li Y. Shortest path problem with multiple shortest paths. *Journal of Harbin Institute of Technology*. 2010. Vol. 42, no. 9. P. 1428–1431.

28. Abraham I., Delling D., Goldber A., Werneck R. A hub-based labeling algorithm for shortest paths in road networks. *Lecture Notes in Computer Science*. 2011. Vol. 6630. P. 230–241. DOI: 10.1007/978-3-642-20662-7_20.

29. Silverbush D., Sharan R. Network orientation via shortest paths. *Bioinformatics*. 2014. Vol. 30, no 10. P. 1449–1455. DOI: 10.1093/bioinformatics/btu043.

30. Ferone D., Festa P., Fugaro S., Pastore T. On the shortest path problems with edge constraints. *Transparent Optical Networks (ICTON'2020)* : Proc. 22nd Int. Conf., Bari, Italy, Jul. 19–23, 2020. P. 1–4. DOI: 10.1109/ICTON51198.2020.9203378.

31. Li J., Wu X. Constrained Shortest Path by Parameter Searching. *Safety Produce Informatization (IICSPI'2019)* : Proc. 2nd Int. Conf., Chongqing, China, Nov. 28–30, 2019. P. 26–29. DOI: 10.1109/IICSPI48186.2019.9095897.

32. Федоренко Г., Фесенко Г., Харченко В. Аналіз методів і розроблення концепції гарантованого виявлення та розпізнавання вибухонебезпечних предметів. *Сучасний стан наукових досліджень та технологій в промисловості*. 2022. № 4 (22). С. 20–31. DOI: 10.30837/ITSSI.2022.21.020.

33. Kharchenko V., Kliushnikov I., Rucinski A., Fesenko H., Illiashenko O. UAV Fleet as a Dependable Service for Smart Cities: Model-Based Assessment and Application. *Smart Cities*. 2022. Vol. 5. Issue 3. P. 1151–1178. DOI: https://doi.org/10.3390/smartcities5030058.

34. Sun Y., Fesenko H., Kharchenko V., Zhong L., Kliushnikov I., Illiashenko O., Morozova O., Sachenko A. UAV and IoT-Based Systems for the Monitoring of Industrial Facilities Using Digital Twins: Methodology, Reliability Models, and Application. *Sensors*. Vol. 22. Issue 17. DOI: https://doi.org/10.3390/s22176444.

# 17. МЕТОДИ І ЗАСОБИ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ НАДІЙНОГО КЕРУВАННЯ МОБІЛЬНИМИ СИСТЕМАМИ ОСВІТЛЕННЯ

**О. А. Чуйко[1], Г. А. Кучук[2]**

[1]***Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»,***
[2]***Національний технічний університет «Харківський політехнічний інститут»***

## 17.1. Вступ

Освітлення є важливим аспектом забезпечення комфортного життя в будь-якому приміщенні. Однак, розумний контроль освітлення може забезпечити більш ефективне використання електроенергії та комфортніші умови проживання. Завдяки штучному інтелекту та інтернету речей (IoT), віддалене керування освітленням може стати значно простішим та ефективнішим. Сучасні мобільні системи освітлення можуть бути здатні не лише пристосовуватися до умов приміщення, але й виконувати інші функції, такі як створення настрою, підвищення продуктивності та забезпечення безпеки.

У даному дослідженні, будуть розглянуті різноманітні методи та засоби штучного інтелекту, які використовуються для надійного керування мобільними системами освітлення. Також у роботі буде проведено огляд сучасних технологій та пристроїв, що дозволяють реалізувати ефективну та економічну доцільність використання цих систем. Дослідження спрямоване на те, щоб дати огляд можливостей, які забезпечуються засобами штучного інтелекту для керування освітленням, та проаналізувати їхні переваги та недоліки.

Також будуть розглянуті такі теми, як аналіз відомих рішень в області мобільних систем освітлення, їхні переваги та недоліки, а також нові рішення та технології, що з'явилися на ринку в останні роки. У роботі також буде розглянуто приклади використання цих систем в різних галузях, таких як домашнє господарство, офісні приміщення та військова сфера.

Отримані результати дозволять зробити висновки про ефективність використання таких систем та їхній вплив на підвищення комфорту проживання та економію електроенергії. У підсумку, дослідження дозволить зрозуміти, як засоби штучного інтелекту можуть забезпечити ефективне та надійне керування мобільними системами освітлення, а також як це може сприяти підвищенню комфорту та безпеки в будь-якому приміщенні, включаючи військові об'єкти.

17.2. Аналіз методів навчання штучного інтелекту з метою інтеграції в систему керування освітленням

Розглянемо такі методи, як навчання з підсиленням, навчання з учителем, навчання без учителя та навчання з використанням глибинного навчання.

Навчання з підсиленням є одним з найпоширеніших методів навчання, який використовується для навчання агентів, що мають здатність приймати рішення на основі взаємодії з навколишнім середовищем. Для цього агент повинен мати можливість спостерігати дії, які він виконує, та отримувати відповіді від середовища на свої дії. Під час навчання з підсиленням агент намагається максимізувати нагороду, яку він отримує від середовища за свої дії. Цей метод може бути ефективним для навчання систем керування освітленням, оскільки вони можуть взаємодіяти з навколишнім середовищем та приймати рішення на основі отриманих від нього даних.

Навчання з учителем полягає в тому, що модель навчається на основі наданої людиною інформації, яка містить відомості про правильні рішення в певній ситуації. Цей метод може бути корисним для навчання систем керування освітленням, якщо нам потрібно знати, які рішення має приймати система в певних ситуаціях.

Навчання без учителя є методом навчання, при якому модель навчається на основі нерозмічених даних. Цей метод може бути корисним для систем керування освітленням, якщо нам потрібно знайти в них залежності та закономірності, не задаючи перед цим конкретних правильних рішень.

Навчання з використанням глибинного навчання полягає в тому, що модель навчається на основі шарів, що представляють різні рівні абстракції даних. Цей метод може бути корисним для навчання систем керування освітленням, якщо ми маємо велику кількість даних, які ми можемо використати для навчання моделі.

Після проведеного аналізу методів навчання штучного інтелекту, ми можемо визначити, який з цих методів може бути найбільш ефективним для навчання систем керування освітленням. Також важливо розглянути методи інтеграції навчання штучного інтелекту в систему керування освітленням, щоб забезпечити їхню надійність та ефективність.

На основі порівняльної таблиці методів навчання штучного інтелекту для інтеграції в систему керування освітленням (табл. 17.1) можна зробити кілька коментарів та висновків:

- кожен метод навчання має свої переваги та недоліки, залежно від конкретних умов та завдань, необхідно обрати найбільш ефективний метод навчання;
- навчання зі зразків може бути дуже ефективним, якщо є достатньо прикладів, однак, цей метод може бути підверженим перенавчанню, тому потрібно бути обережним при його використанні;

- навчання з підсиленням є ефективним у випадках, коли немає можливості використовувати попередньо підготовлені приклади, однак, цей метод вимагає багато часу та ресурсів для навчання;
- глибинне навчання є досить складним методом навчання, але він забезпечує високу точність результатів, проте, для його використання потрібно мати значні обчислювальні ресурси та експертизу;
- навчання без вчителя може бути ефективним у випадках, коли немає попередньо підготовлених даних, однак, результати можуть бути важко інтерпретувати.

Таблиця 17.1 – Порівняльна таблиця методів навчання штучного інтелекту

| Метод | Опис | Переваги | Недоліки |
|---|---|---|---|
| Навчання зі зразків | Модель навчається на основі попередньо підготовлених прикладів | Простота інтерпретації результатів, швидкість навчання | Потребує велику кількість даних, може бути уразливий перенавчанню |
| Навчання з підсиленням | Модель навчається на основі взаємодії з навколишнім середовищем | Ефективний у випадках, коли не можна використовувати зразки | Вимагає багато часу та ресурсів для навчання |
| Глибинне навчання | Модель навчається на основі різних рівнів абстракцій даних | Висока точність результатів | Вимагає значних обчислювальних ресурсів та експертизи |
| Навчання без вчителя | Модель навчається на основі не маркованих даних | Може знайти складні залежності та закономірності | Результати можуть бути важко інтерпретувати |

Отже, для досягнення найкращих результатів у системі керування освітленням, необхідно ретельно обирати метод навчання штучного інтелекту відповідно до конкретної задачі та наявних даних.

На основі порівняльної таблиці можна зробити висновок, що для задачі керування освітленням та її використання в військовій галузі найбільш підходять методи навчання з підсиленням та еволюційне навчання.

Метод навчання з підсиленням дозволяє системі самостійно вивчати, які дії є оптимальними в різних ситуаціях, з урахуванням взаємодії з довкіллям. Це

особливо важливо в військовій галузі, де можуть виникнути непередбачувані ситуації, які потребують швидкого та ефективного рішення.

Еволюційне навчання також може бути корисним, оскільки воно дозволяє системі самостійно змінюватись та адаптуватись до довкілля. Це може бути корисним у військовій галузі, де швидкі зміни в ситуації можуть вимагати швидкої зміни стратегії.

Отже, методи навчання з підсиленням та еволюційне навчання можуть бути ефективними для використання в системах керування освітленням та військовій галузі. Однак, перед використанням будь-якого методу, необхідно ретельно вивчити всі його переваги та недоліки, щоб визначити, який метод найбільш відповідає потребам конкретної задачі.

## 17.3. Аналіз випадків інтеграції і використання штучного інтелекту у військовій галузі

В останні роки штучний інтелект став необхідним елементом військової техніки та обладнання. Його можна використовувати в різних сферах, від збору та обробки інформації до автоматичного керування системами зброї та бойової техніки.

Один з прикладів використання штучного інтелекту військовій галузі - це система автоматичного наведення та стрільби, що використовує алгоритми штучного інтелекту для розпізнавання цілей та розрахунку параметрів стрільби.

Інший приклад - це використання алгоритмів машинного навчання для автоматичного визначення місцезнаходження та класифікації цілей на основі даних з різних датчиків та камер. Штучний інтелект також може використовуватися для прогнозування поведінки ворожих сил та виявлення підривних дій на передовій.

Всі ці приклади демонструють ефективність та потенціал використання штучного інтелекту у військовій галузі. Однак, також потрібно враховувати потенційні ризики, пов'язані з використанням штучного інтелекту військовими силами, тому їх використання потребує ретельного аналізу та регулювання.

Ще одним прикладом використання штучного інтелекту військовими є система ELSA (англ. Enhanced Logistics and Supply Architecture). Ця система використовується для підтримки поставок зброї та іншого військового спорядження. Вона забезпечує постачальників військового спорядження інформацією про стан запасів та вимоги до поставок в режимі реального часу.

Наступним прикладом використання штучного інтелекту у військовій галузі є система бойових дронів. Один з найбільш відомих прикладів використання штучного інтелекту військовими силами - це система керування дронами. Наприклад, система MQ-9 Reaper (рис. 17.1), яку використовують в армії США, включає в себе комп'ютерну програму, яка дозволяє дрону автоматично літати, виявляти та ідентифікувати цілі, проводити наведення зброї та здійснювати маневри на основі отриманих даних.

Ця система має дуже високу точність та швидкість реакції, що дозволяє військовим здійснювати оперативні дії без ризику для людського життя. Крім

того, використання дронів зі штучним інтелектом дозволяє військовим отримувати детальну та точну інформацію про територію, що сприяє більш ефективній плануванню та виконанню операцій.

Такі системи забезпечують військовим можливість проведення війни на відстані, що дозволяє захистити власних солдат від прямого вогню противника. Застосування штучного інтелекту військовими дозволяє значно підвищити ефективність виконання завдань та зменшити ризик для людей, що може мати ключове значення в сучасних конфліктах.

Рисунок 17.1 – Бойовий БПЛА MQ-9 Reaper

Інший приклад використання штучного інтелекту у військовій галузі - це програма передбачення потреб військових бюджетів. Вона використовує аналітичні моделі, щоб прогнозувати витрати на різні типи військових операцій та оцінювати, як різні фактори впливатимуть на витрати.

В цьому розділі ми проаналізували декілька реальних випадків використання штучного інтелекту військовими (табл. 17.2). Ці приклади демонструють, що штучний інтелект може бути використаний в різних сферах військової діяльності, від поставок військового спорядження до розвідки та бойових дій.

Таблиця 17.2 – Сфери використання та інтеграції ШІ у військовій галузі:

| Галузь | Приклад використання штучного інтелекту |
|---|---|
| Дрони та БПЛА | ¬ Автоматичне планування маршрутів;<br>¬ Розпізнавання ворожої армії<br>¬ Розпізнавання ворожої техніки |
| Автомобілі | ¬ Автопілот<br>¬ Система попередження при зіткненні |
| Озброєння | ¬ Розпізнавання цілей<br>¬ Автоматичне наведення<br>¬ Автоматизація системи пусків та прийняття рішень |
| Кібербезпека | ¬ Виявлення вразливостей в мережах<br>¬ Аналіз потенційних загроз |

## 17.4. Порівняння варіантів використання інтелектуальних систем освітлення

У цьому розділі ми розглянемо порівняння різних варіантів використання інтелектуальних систем освітлення.

Один з можливих варіантів - це використання системи з датчиками руху, яка буде автоматично включати світильники в приміщенні, коли хтось з'являється в зоні дії датчика, і вимикати їх, коли нікого немає.

Інший варіант - використання системи із застосуванням камер та аналізу зображень. Ця система може визначати кількість людей в приміщенні та їх розташування, і забезпечувати належне освітлення в залежності від цього.

Третій варіант - це використання системи, яка може регулювати освітлення в залежності від часу доби та погодних умов. Наприклад, вранці та ввечері, коли є менше світла, система автоматично збільшує яскравість світильників, а вдень, коли світло досить яскраве, система зменшує їх яскравість для економії енергії.

Останній варіант - використання системи, яка забезпечує індивідуальний підхід до кожного користувача. Наприклад, система може використовувати датчики освітлення, щоб визначити, які світильники потрібні на основі віку та здоров'я користувача.

Кожен з цих варіантів має свої переваги та недоліки, і вибір підходу залежить від конкретних потреб та обмежень.

Розглянемо детальніше реалізацію кожного варіанту використання інтелектуальних систем освітлення:

Освітлення на основі сенсорів руху та присутності людини:

Цей варіант включає в себе встановлення ряду сенсорів руху та присутності людини, які забезпечують автоматичне включення світла, коли людина знаходиться в приміщенні, а також автоматичне вимкнення світла, якщо приміщення порожнє. Для реалізації такого варіанту використовуються датчики руху, інфрачервоні сенсори та інші пристрої, які реагують на присутність людини в приміщенні.

### 17.4.1. Освітлення на основі програмного керування

Цей варіант передбачає використання програмного забезпечення для керування освітленням. За допомогою програмного забезпечення можна створювати різні сценарії освітлення, регулювати яскравість світла, встановлювати таймери тощо. Для реалізації такого варіанту використовуються комп'ютери, контролери, програмні засоби та інші пристрої.

### 17.4.2. Освітлення на основі штучного інтелекту

Цей варіант включає в себе використання систем штучного інтелекту для керування освітленням. Системи штучного інтелекту можуть аналізувати дані про присутність людини, освітлення, час доби, погодні умови та інші параметри

для автоматичного керування освітленням. Для реалізації такого варіанту використовуються спеціалізовані системи штучного інтелекту, датчики руху та присутності (табл. 17.3), інтернет-підключення та інші пристрої

Таблиця 17.3 – Опис типу датчиків для системи освітлення та варіанти їх використання

| Тип датчику в залежності від системи | Опис | Використання у військовій галузі | Місце встановлення |
|---|---|---|---|
| Датчик руху | Система включає світло, коли відчуває рух у приміщенні та вимикає його, коли руху немає. | Застосування відслідковування руху на полі бою. | Поле бою, склади, місця розташування техніки. |
| Датчик присутності | Система включає світло, коли відчуває наявність людини у приміщенні та вимикає його, коли нікого немає. | Моніторинг присутності людей на військових об'єктах. | Контрольний пункт, магазини зброї, місця зберігання важливої інформації. |
| Система контролю освітленості | Система вимірює рівень освітленості та включає або вимикає світло, щоб забезпечити оптимальний рівень освітленості. | Застосування на збройних частинах для забезпечення оптимальної освітленості. | Збройні частини, командні пункти, медичні заклади. |

17.4.3. Порівняння інтелектуальних систем освітлення

У табл. 17.4 порівнюються чотири варіанти використання інтелектуальних систем освітлення, їх переваги та недоліки.

За результатами аналізу можна зробити висновок, що система штучного інтелекту має найбільші переваги, однак, така система може бути надто складною для більшості простих приміщень та недосяжною з фінансової точки зору.

В таких випадках можна розглянути використання датчика освітленості або датчика руху як альтернативу. Таймер може бути дешевим варіантом, але не може дати користувачеві повного контролю над освітленням.

Таблиця 17.4 – Недоліки та переваги існуючих систем у порівнянні із системою під керуванням ШІ

| Тип системи | Переваги | Недоліки |
|---|---|---|
| Система простого таймеру | Низька вартість, проста установка | Неможливість автоматичного регулювання освітлення залежно від зовнішніх умов та потреб користувача |
| Система на датчиках руху | Автоматичне включення світла при вході у приміщення, економія електроенергії | Недостатньо точне визначення наявності людей у приміщенні, можливі помилкові включення |
| Система на базі датчика освітленості | Автоматичне регулювання освітлення в залежності від зовнішніх умов, економія електроенергії | Недостатньо точне визначення рівня освітленості, можливі помилкові регулювання |
| Система штучного інтелекту | Можливість автоматичного регулювання освітлення залежно від зовнішніх умов та потреб користувача, висока точність, можливість аналізу даних та вдосконалення системи | Висока вартість, складність установки та програмування |

Таким чином, інтелектуальні системи освітлення можуть бути реалізовані за допомогою різних методів і технологій. Кожен з цих методів має свої переваги та недоліки, які повинні бути враховані при виборі конкретного варіанту. Наприклад, варіант з використанням навчальних даних, зібраних за допомогою датчиків руху та світла, може бути ефективним для використання в місцях, де велика кількість людей переміщується протягом дня, таких як торгові центри, аеропорти або вокзали. Водночас, цей метод може виявитися неефективним для використання у приватних домах, де режим освітлення залежить від розкладу сімейних обідів та звичок кожного члена родини.

Варіант з використанням алгоритмів машинного навчання та нейронних мереж може бути ефективним для використання в широкому спектрі умов, оскільки він здатний адаптуватися до змінних умов та працювати з різноманітними типами даних. Однак, для реалізації цього варіанту необхідно

мати достатню кількість навчальних даних та витратити час та зусилля на їх збір та обробку.

Окремим варіантом є використання системи з контролем голосу, яка може бути ефективною в місцях, де користувачі не можуть взаємодіяти з системою безпосередньо, наприклад, під час виконання військових операцій або в лікарнях, де пацієнти мають обмежені можливості руху.

## 17.5. Впровадження комп'ютерних систем керування освітленням

В результаті аналізу, було розроблено алгоритм та послідовність впровадження комп'ютерних систем керування освітленням у середовище. Нижче наведено послідовність, етапи алгоритму впровадження, та пояснення кожного етапу інтеграції.

1. Вибір системи керування освітленням

Першим кроком є вибір системи керування освітленням, яка буде використовуватися. Важливо враховувати такі параметри, як тип системи (комп'ютерна або не комп'ютерна), вартість, можливості налаштування, розмір і склад приміщення, технічні вимоги, можливість інтеграції з іншими системами.

2. Підготовка інфраструктури для встановлення системи

Після вибору системи керування освітленням необхідно підготувати інфраструктуру для її встановлення. Це може включати в себе планування розташування світильників, прокладання електропроводки, розробку схеми підключення системи керування.

3. Встановлення системи

Після підготовки інфраструктури необхідно встановити систему керування освітленням. Це може вимагати використання додаткового обладнання або програмного забезпечення для підключення системи до мережі електроживлення.

4. Налаштування системи

Після встановлення системи необхідно налаштувати її роботу. Це включає в себе встановлення параметрів освітлення, налаштування сценаріїв освітлення, додавання нових пристроїв до системи та інші дії, які дозволяють налаштувати систему на потрібний режим роботи.

5. Тестування і перевірка роботи системи

Після налаштування системи необхідно провести тестування та перевірку її роботи. Це дозволяє виявити можливі проблеми та помилки, які потребують додаткової настройки або виправлення.

6. Експлуатація системи і забезпечення безперебійної роботи

Після успішного тестування система готова до експлуатації. Необхідно встановити режим роботи системи та забезпечити її безперебійну роботу. Для цього необхідно проводити регулярне технічне обслуговування та вчасно вирішувати можливі проблеми.

7. Підтримка інтеграції з іншими системами

Система керування освітленням може бути інтегрована з іншими системами, такими як система керування кліматом, система безпеки та інші. Для забезпечення коректної роботи такої інтеграції потрібно провести підтримку та налаштування системи на цей процес.

8. Оцінка ефективності системи

Оцінка ефективності системи керування освітленням дозволяє визначити, наскільки ефективно вона виконує свої функції та скільки коштує експлуатація. Це дозволяє планувати подальші дії з покращення роботи системи та зменшення її експлуатаційних витрат.

Після оцінки ефективності можна вирішувати, чи потрібні додаткові покращення та модифікації системи, або якщо система працює належним чином, можна розглянути можливість розширення її функціоналу або інтеграції з іншими системами.

Важливо мати на увазі, що система керування освітленням є лише одним з елементів автоматизації будинку, виробництва або офісу, і для досягнення максимальної ефективності рекомендується інтегрувати її разом з іншими системами керування.

## 17.6. Огляд існуючих типів та видів освітлювальних приладів під керуванням протоколу DMX

Протокол DMX (Digital Multiplex) є одним з найпопулярніших протоколів для керування освітленням. Він дозволяє передавати цифрові сигнали для керування різними типами освітлювальних приладів, таких як світлодіодні прожектори, прилади з рухомими головками, LED-смужки та інші. Крім того, протокол DMX може керувати такими приладами, як струмоприймачі, реле, димомашини, стабілізатори напруги та інші прилади для керування освітленням.

Розглянемо найпопулярніші типи приладів освітлення:

1. Світлодіодні прожектори – ці прилади використовують світлодіоди для створення світла (рис. 17.2). Вони можуть мати різну потужність і кольорову гаму.

Рисунок 17.2 – Світлодіодний прожектор

2. Прилади з рухомими головками (рис. 17.3) – ці прилади мають головки, які можуть рухатися в різних напрямках. Вони дозволяють створювати різні ефекти, наприклад, кружляючі лінії світла або малюнки, що рухаються.

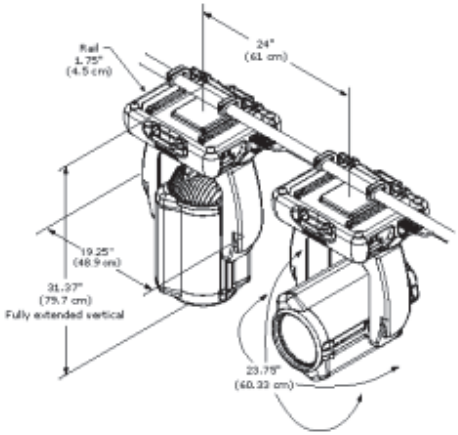



Рисунок 17.3 – Прилад з рухомими головками

3. LED-смужки (рис. 17.4) – ці прилади складаються з ряду світлодіодів, які можуть світитися в різних кольорах. Вони дозволяють створювати ефекти освітлення, такі як м'які переходи між кольорами або смуги світла, що рухаються.

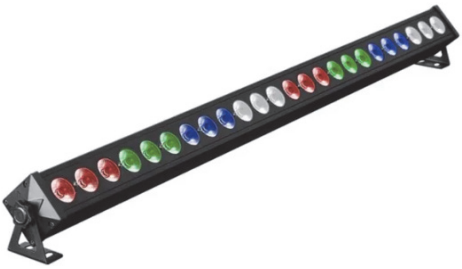


Рисунок 17.4 – LED-смужка

4. Прилади з керованими діаметрами прожекторів (рис. 17.5) – ці прилади дозволяють змінювати діаметр світлового променю. Вони зазвичай використовуються для створення різних ефектів освітлення, наприклад, для підсвічування конкретних деталей.

Рисунок 17.5 – Прилад з керованими діаметрами прожекторів

За допомогою протоколу DMX можна керувати як окремими приладами, так і цілими системами освітлення. Наприклад, з його допомогою можна керувати освітленням в театрі, на концерті, у нічному клубі або в іншому місці з великим обсягом освітлення. Протокол DMX дозволяє передавати до 512 каналів керування в одному кабелі, що дозволяє значно спростити процес керування освітленням та зменшити кількість кабелів, необхідних для керування всіма приладами. Також це дозволяє забезпечити точне керування яскравістю, кольором та іншими параметрами світла.

Поєднання штучного інтелекту та протоколу DMX має потенціал дозволити створення більш складних та ефективних систем керування освітленням. Наприклад, штучний інтелект може допомогти забезпечити більш точну та швидку керованість приладами, що підтримують протокол DMX, забезпечуючи кращу реакцію на зміни вимог та умов.

Додатково, штучний інтелект може допомогти в розробці більш інтелектуальних та автоматизованих систем керування освітленням, які будуть можливі завдяки підтримці протоколу DMX. Наприклад, системи можуть автоматично адаптуватися до змін вимог до освітлення в залежності від різних факторів, таких як пори року, час доби, погода та наявність людей в приміщенні.

Крім того, використання штучного інтелекту може допомогти в оптимізації споживання енергії та зниження витрат на опалення та освітлення. Системи керування, які комбінують протокол DMX та штучний інтелект, можуть аналізувати використання електроенергії та прогнозувати її витрати, забезпечуючи більш ефективне та економічне використання ресурсів.

Загалом, поєднання штучного інтелекту та протоколу DMX відкриває широкі можливості для розвитку нових технологій та систем керування освітленням, що можуть забезпечити більш ефективне та інтелектуальне використання енергії, більш точну та швидку реакцію на зміни умов та більш ефективне використання ресурсів.

## 17.7. Висновки

На основі проведеного у цій роботі аналізу та досліджень можна зробити висновок, що перспективи розвитку надійних систем керування освітленням за допомогою штучного інтелекту в військовій галузі є досить високими. Застосування штучного інтелекту у таких системах може значно покращити якість та ефективність роботи, знизити ризики помилок та забезпечити високий рівень безпеки. Використання різноманітних датчиків, які можуть зібрати інформацію про стан довкілля та об'єктів, також може значно поліпшити якість та точність роботи системи.

Протокол DMX забезпечує зручне та ефективне керування різними типами освітлювальних приладів, що значно розширює можливості використання таких систем в військовій галузі. Крім того, використання мережевих технологій дозволяє забезпечити швидку та надійну передачу даних, що є особливо важливим у військових умовах. Однак, розробка та впровадження надійних систем керування освітленням за допомогою штучного інтелекту в військовій галузі потребує великих зусиль та інвестицій. Для досягнення успіху в цьому напрямку необхідно провести докладні дослідження, розробити та впровадити нові технології, а також забезпечити високий рівень кваліфікації персоналу.

Отже, можна стверджувати, що перспективи розвитку надійних систем керування освітленням за допомогою штучного інтелекту в військовій галузі є досить високими, але для досягнення успіху в цьому напрямку потрібно вкласти великі зусилля та інвестиції. Однак, враховуючи потенційні переваги, такі як покращення ефективності та забезпечення безпеки, розвиток цих систем може бути важливим кроком у покращенні військової діяльності та забезпеченні високої готовності до дій в умовах будь-якої складності.

Подальші дослідження у галузі надійних систем керування освітленням за допомогою штучного інтелекту в військовій галузі можуть включати більш глибокий аналіз та дослідження різноманітних методів і технологій, які можуть бути використані для розробки таких систем.

Дослідження можуть охоплювати розробку нових алгоритмів машинного навчання та глибинного навчання для забезпечення більш точного та швидкого реагування системи на зміни в довкіллі та ситуації на полі бою. Також можуть досліджуватися можливості використання додаткових датчиків та джерел інформації, які можуть допомогти збирати більше даних про стан довкілля та об'єктів.

Додатково, дослідження можуть включати розробку нових технологій зв'язку та передачі даних, які можуть забезпечити ще більшу надійність та швидкість передачі інформації. Також можуть проводитися дослідження у галузі енергоефективності та енергозбереження, що може допомогти знизити споживання енергії системою та забезпечити більш тривалий час роботи в умовах, коли життєво важливі ресурси обмежено.

Усі ці дослідження можуть допомогти розробити та впровадити більш надійні та ефективні системи керування освітленням за допомогою штучного

інтелекту в військовій галузі, що може значно покращити безпеку та ефективність дій військових підрозділів.

Література


1. Van Harmelen, F., ten Teije, A., “Compositional patterns for combining KR & ML: a first attempt,” in Pre-Proceedings of the Cognitive Computation Symposium: Thinking Beyond Deep Learning, (2018).
2. Van Harmelen, Frank, and Annette ten Teije, “A Boxology of Design Patterns for Hybrid Learning and Reasoning Systems,” arXiv preprint arXiv:1905.12389, (2019).
3. Serafini, L., & Garcez, A. D. A., “Logic tensor networks: Deep learning and logical reasoning from data and knowledge,” arXiv preprint arXiv: 1606.04422, (2016).
4. Maedche, A., & Staab, S., “Ontology learning. In Handbook on ontologies,” 173 –190 Springer, Berlin, Heidelberg (2004).
5. G. Burghouts, “V1508: Full-Motion Video for Intelligence, Surveillance and Reconnaissance – Description of Algorithms for Metadata Extraction,” (2018).
6. YouTube channel:Anna News, https://www.youtube.com/channel/UCGib-bLlq8HTRp2YaEESxeg
7. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in In European Conference on Computer Vision, 21 –37 (2016). https://doi.org/10.1007/978-3-319-46448-0_2
8. Stap, N., van Opbroek, A. G., Huizinga, W., Wilmer, M. M. G., van den Broek, S. P., Pruim, R. H. R., … & Dijk, J., “Maritime detection framework 2.0: A new approach of maritime target detection in electro-optical sensors,” in Proceedings of SPIE-The International Society for Optical Engineering, (2018).
9. “MIT, Port of Single Shot MultiBox Detector to Keras,” (2017) https://github.com/rykov8/ssd_keras
10. Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A., “The PASCAL Visual Object Classes Challenge (VOC2007),” http://www.pascal-network.org/challenges/VOC/voc2007/index.html
11. Van Ramshorst, A., “Automatic Segmentation of Ships in Digital Images: A Deep Learning Approach,” (2018).
12. N. van der Stap, J. Dijk, “V1423 MEOSS demonstration,” (2018).
13. Bouma, H., Schutte, K., ten Hove, J. M., Burghouts, G., & Baan, J., “Flexible human-definable automatic behavior analysis for suspicious activity detection in surveillance cameras to protect critical infrastructures,” Counterterrorism, Crime Fighting, Forensics, and Surveillance Technologies II, 10802 108020N International Society for Optics and Photonics.2018).
14. Schutte, K., Burghouts, G., van der Stap, N., Westerwoudt, V., Bouma, H., Kruithof, M., … & ten Hove, J. M., “Long-term behavior understanding based on the expert-based combination of short-term observations in high-resolution CCTV,” Optics and Photonics for Counterterrorism, Crime Fighting, and Defence XII, 9995 99950Q International Society for Optics and Photonics.2016).

15. Analysis of Artificial Neural Network Architectures for Modeling Smart Lighting Systems for Energy Savings - Alberto Garcés-Jiménez, José Luis Castillo-Sequera, Antonio Del Corte-Valiente, José Manuel Gómez-Pulido, Esteban Patricio Domínguez González-Seco;

16. Artificial Neural Network Based Power Management for Smart Street Lighting Systems -Dr. S. Smys, Dr. Abul Basar, Dr. Haoxiang Wang;

17. Smart Lighting Application for Energy Saving and User Well-Being in the Residential Environment - Moe Soheilian, Géza Fischl, Myriam Aries;

18. Computer-Assisted Lighting Design and Control - Dipl.-Inform. Michael Sperber;

19. Sah, S. Machine Learning: A Review of Learning Types. Preprints 2020, 2020070230 (doi: 10.20944/preprints 202007. 0230.v1).

20. GrandMA online manual - https://www.malighting.com/training-support/ma-university/overview/

21. Jinjiang Hui. Design of intelligent lighting energy-saving control system based on LED [J]. Engineering Construction and Design, 2020, (14): 253-254.

22. J. Purmaissur, A. Seeam, S. Guness and X. Bellekens, "Augmented Reality Intelligent Lighting Smart Spaces," 2019 Conference on Next Generation Computing Applications (NextComp), 2019, pp. 1-5, doi: 10.1109/NEXTCOMP.2019.8883577.

23. Bartneck, C., Lütge, C., Wagner, A., Welsh, S. (2021). Military Uses of AI. In: An Introduction to Ethics in Robotics and AI. SpringerBriefs in Ethics. Springer, Cham. https://doi.org/10.1007/978-3-030-51110-4_11

24. Ronald Arkin. Governing lethal behavior in autonomous robots. Chapman and Hall/CRC, 2009. ISBN 978-1420085945. URL http://www.worldcat.org/oclc/933597288

25. E. Hwang, "Smart lighting systems: An overview of energy-efficient lighting and advanced control systems," Renewable and Sustainable Energy Reviews, vol. 45, pp. 1-16, Jan. 2015.

26. S. S. Sivaji and M. Pandikumar, "A review of intelligent lighting control systems for energy-efficient buildings," Journal of Building Engineering, vol. 36, pp. 101781, Jul. 2021.

27. M. A. Al-Qassab and M. A. Al-Mosawi, "Smart Lighting System Based on IoT Using Raspberry Pi and Arduino," Journal of Engineering, vol. 2020, pp. 1-11, Jan. 2020.

28. C. K. Ho, K. W. Chan, and J. H. Li, "Integration of DMX512 and ZigBee networks for intelligent lighting control," International Journal of Advanced Computer Science and Applications, vol. 8, no. 4, pp. 128-134, Apr. 2017.

29. D. D. Dzung and L. D. Tuan, "An advanced wireless control system for lighting based on DMX-512 protocol," International Journal of Engineering and Advanced Technology, vol. 8, no. 4, pp. 87-93, Apr. 2019.

30. J. Xue, W. Wang, and Y. Liu, "Intelligent lighting control system based on ZigBee wireless sensor network," in 2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, 2015, pp. 210-213.

31. H. F. Liao, T. L. Chiu, and C. C. Chien, "Development of a smart lighting control system using occupancy sensors and wireless communication technology," Energy Procedia, vol. 62, pp. 331-337, Oct. 2014.

32. J. O. Pedro and P. S. Girão, "Towards energy-efficient buildings: A survey of lighting control technologies," Energy and Buildings, vol. 170, pp. 160-177, Nov. 2018.

33. J. Zhang, Z. Guo, and Z. Chen, "Research on intelligent control system of lighting based on computer vision technology," in 2018 14th IEEE International Conference on Electronic Measurement & Instruments, 2018, pp. 736-741.

34. Y. T. Lin, K. C. Chen, and C. W. Huang, "Design and implementation of an intelligent lighting control system with energy-saving strategies for indoor applications," Applied Sciences, vol. 9, no. 17, pp. 3501, Aug. 2019.

# 18. ЕФЕКТИВНІСТЬ ВИКОРИСТАННЯ ЯВНИХ ТА НЕЯВНИХ КОРИСТУВАЦЬКИХ ФІДБЕКІВ ДЛЯ ГІБРИДНИХ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

**А. І. Кулягін[1], Г. А. Кучук[2]**

*[1]Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»*
*[2]Національний технічний університет «Харківський політехнічний інститут»*

## 18.1. Вступ

Рекомендаційні системи відіграють важливу роль у спрощенні пошуку та відбору релевантного контенту та продуктів для користувачів. Вони аналізують інтереси користувачів і переглянутий контент для формування персоналізованих рекомендацій. Визначення ефективності рекомендацій в таких системах все ще залишається відкритим питанням, особливо коли йдеться про використання явних (наприклад, оцінки та відгуки користувачів) та неявних (історія переглядів, час перегляду, взаємодія з контентом – кліки, пауза, навігація) фідбеків користувачів [1]. Проблематика полягає в необхідності урахування відмінностей поведінки користувачів, оцінці значимості кожного виду фідбеку та використанні різних видів фідбеків для досягнення найкращих результатів.

Для покращення рекомендацій системи, зокрема, виявлення неочевидних інтересів користувачів, підвищення охоплення продуктів, збільшення переглядів товарів, тощо, використовуються різні підходи. Зокрема це розробка адаптивних алгоритмів [2], які допомагають виявити зміни уподобань користувачів на основі явних та неявних фідбеків, виявлення та усунення можливих упереджень в явних оцінках користувачів, використовуючи неявні фідбеки.

Метою цього дослідження є аналіз методів визначення ефективності впливу явних та неявних користувацьких фідбеків на результати рекомендацій в гібридних рекомендаційних системах. Також, дане дослідження спрямоване на пошук нових підходів до використання явних та неявних фідбеків для підвищення якості рекомендацій, зокрема знаходження нових та прихованих інтересів користувачів.

## 18.2. Метрики для оцінки ефективності явних та неявних фідбеків

Щоб оцінити вплив явних та неявних фідбеків від користувачів гібридних рекомендаційних систем, необхідно визначити метрики, що дозволяють оцінити результати рекомендацій. У цьому розділі буде розглянуто основні

види явних та неявних фідбеків та метрики, які можна використовувати для оцінки їх ефективності.

Явні фідбеки можуть включати:

1. Явна оцінка продуктів або контенту за якоюсь шкалою.
2. Відгуки – текстові відгуки або коментарі про продукти або контент.
3. Вподобання або не вподобання.

Неявні фідбеки:

1. Історія переглядів – перелік продуктів або контенту, який користувач переглянув.
2. Тривалість перегляду контенту користувачем.
3. Кількість натискань користувача на ті чи інші рекомендації.
4. Частота перегляду окремих продуктів або контенту, перегляд схожих за тематикою рекомендацій.
5. Додавання до списку бажань або кошика, зберігання користувачем контенту.

Існує кілька показників для оцінки моделей рекомендаційних систем [3]. Вибір кожного з них ґрунтується на типі рекомендаційної системи та типі фідбеків. Якщо це робота з рекомендаціями на основі контенту, необхідно обирати з метрик подібності. Для колаборативної фільтрації необхідно використовувати метрики передбачення (якщо мова йде про оціночне передбачення) та метрики класифікації (для бінарного передбачення). Коли ж мова йде про гібридні рекомендаційні системи – необхідно враховувати обидва підходи та надавати їм різні ваги значимості.

Серед прогнозованих метрик оцінки ефективності фідбеків можна виділити такий перелік:

1. MAE (Середня абсолютна похибка) - вимірює середню абсолютну різницю між фактичним та передбаченим результатом. Ця метрика важлива, оскільки допомагає оцінити, наскільки точно рекомендаційна система передбачає оцінки користувачів. Чим нижче значення MAE, тим точніше передбачення системи. MAE дорівнює середньому значенню абсолютних різниць між фактичними оцінками користувачів та передбаченнями системи:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - p_i|$$

де N - кількість оцінок, $y_i$ - фактична оцінка користувача, $p_i$ - передбачена оцінка системи.

2. RMSE (Коренева середня квадратична помилка) – дозволяє виміряти середньоквадратичну різницю між фактичними та передбаченими оцінками користувачів. Важливість RMSE полягає в тому, що вона штрафує великі помилки сильніше, ніж малі. Це означає, що системи з низьким значенням RMSE краще працюють з точки зору передбачення оцінок користувачів, особливо в контексті явних фідбеків:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - p_i)^2}$$

де N - кількість оцінок, $y_i$ - фактична оцінка користувача, $p_i$ - передбачена оцінка системи.

Класифікаційні метрики оцінюють здатність рекомендаційних систем приймати рішення. Такі метрики є хорошим вибором для таких завдань, як визначення релевантних або нерелевантних продуктів для користувача. Для метрик підтримки прийняття рішень точний рейтинг ігнорується, тоді як для методів на основі ранжирування він має неявний вплив через рейтинг. Серед класифікаційних метрик можна виділити наступні:

1. Precision (Точність) – відсоток релевантних рекомендацій серед усіх рекомендацій, наданих системою. Важливість точності полягає в тому, що вона оцінює якість рекомендацій, що надаються користувачам. Чим вище точність, тим менше нерелевантних рекомендацій отримують користувачі, що сприяє задоволеності користувачів та вірності бренду. Формула для отримання точності:

$$Precision = \frac{TP}{TP + FP}$$

де TP - кількість істинно позитивних рекомендацій (релевантні рекомендації, вірно визначені системою), FP - кількість хибно позитивних рекомендацій (нерелевантні рекомендації, помилково визначені системою як релевантні).

2. Recall (Повнота) – це частка найпопулярніших рекомендованих елементів, які є в наборі елементів, релевантних для користувача. Повнота важлива для максимізації знаходження релевантних рекомендацій для користувачів. Чим більше повнота, тим вищий коефіцієнт попадання, оскільки ймовірність того, що правильна відповідь буде розглянута в рекомендаціях більше. Формула для повноти:

$$Recall = \frac{TP}{TP + FN}$$

де TP - кількість істинно позитивних рекомендацій (релевантні рекомендації, вірно визначені системою), FN - кількість хибно негативних рекомендацій (релевантні рекомендації, помилково визначені системою як нерелевантні).

3. F1-Score – це гармонійне середнє значення точності та повноти, яке допомагає об’єднати їх у одну метрику. Цей коефіцієнт не враховує істинно-від'ємні значення. Це ті випадки, коли система рекомендацій не рекомендувала нерелевантний для користувача товар. Це означає, що можна встановити будь-

яке значення проти істинно-від'ємних значень, і це не вплине на результат F1-Score:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

4. MCC (Коефіцієнт кореляції Метьюза) – це коефіцієнт кореляції між спостережуваною та прогнозованою бінарною класифікацією. Коли класифікатор ідеальний (FP = FN = 0), значення MCC дорівнює 1, що вказує на ідеальну позитивну кореляцію. І навпаки, коли класифікатор завжди неправильно класифікує (TP = TN = 0), ми отримуємо значення -1, що представляє ідеальну негативну кореляцію:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}}$$

Якщо є алгоритм генерації кандидатів, який повертає ранжований порядок елементів, а елементи нижче в списку менш імовірно будуть використані або побачені, тоді слід враховувати наступні показники:

1. Average precision (середня точність) – дозволяє розглянути лише підмножину рекомендацій з рангом від 1 до k. Середня точність винагороджує систему за розміщення правильних рекомендацій на початку списку:

$$AP = \frac{1}{m} \sum_{k=1}^{N} (P(k) \text{ if } k^{th} \text{ item was relevant})$$

де N - кількість елементів для рекомендації, m – кількість релевантних елементів, $p(k)$ – передбачена релевантна оцінка системи.

2. MAP (Mean average precision) – на відміну від Average precision, яка застосовується до однієї точки даних, що еквівалентно одному користувачеві, MAP є середнім показником Average precision для всіх користувачів Q:

$$MAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q}$$

3. ARHR (середній взаємний ранг попадання) або MRR (середній взаємний ранг) – це середнє значення взаємного рангу (RR) для користувачів. Зворотний ранг є «множинною зворотною» рангу першого правильного елемента. MRR доречний коли є лише один відповідний пункт або коли лише перший рекомендований елемент є основним. Це означає, що MRR не застосовується, якщо в отриманому списку є кілька правильних відповідей. Формула MRR виглядає наступним чином:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i}$$

Вибір даних метрик допомагає отримати загальне уявлення про ефективність явних та неявних фідбеків [4], а також про те, як ці фідбеки впливають на роботу системи. Використовуючи ці метрики, можна виявити можливі проблеми та вдосконалити рекомендаційні системи, щоб краще задовольняти потреби користувачів та забезпечувати високу якість рекомендацій [5].

## 18.3. Аналіз існуючих рішень з використанням явних та неявних видів фідбеку для рекомендаційних систем

Явні фідбеки забезпечують безпосередню інформацію про вподобання користувачів, як-то оцінки або відгуки. Такі фідбеки можуть бути враховані в колаборативній та контент-орієнтованій фільтрації для створення рекомендацій, які відповідають безпосереднім інтересам користувачів. Однак явні фідбеки можуть бути обмежені кількістю даних, оскільки користувачі можуть не надавати оцінки чи відгуки для більшості продуктів або послуг. Інша проблема – надання упереджених оцінок або надто відмінна поведінка для різних груп користувачів. Для того, щоб вирішувати ті чи інші неоднорідності в даних фідбеків, застосовуються різні підходи, зокрема: збільшення масиву даних, застосування нейронних мереж, різні підходи до визначення точності колаборативної фільтрації та контентно-орієнтованих систем.

Неявні фідбеки надають інформацію про поведінку користувачів, таку як перегляди контенту чи товарів, час перегляду, кількість натискань, зони натискань та інші дії. Ці фідбеки можуть допомогти виявити зміни вподобань користувачів, навіть за відсутності явних фідбеків [6]. Проте неявні фідбеки можуть бути менш точними у відображенні вподобань користувачів, оскільки деякі дії можуть не відповідати дійсним інтересам користувача.

Вибір виду фідбеку залежить від контексту застосування рекомендаційної системи, доступності даних та специфіки продуктів або послуг. Враховуючи наступні фактори, можна визначити, які види фідбеків краще використовувати:

1. Доступність даних – коли користувачі активно надають явні фідбеки, такі як оцінки та відгуки, рекомендаційна система може використовувати такі дані для створення більш точних та особистих рекомендацій. Проте, якщо явних фідбеків недостатньо або вони відсутні, система може використовувати неявні фідбеки, такі як історія переглядів та дії користувачів, для створення рекомендацій.
2. Динаміка вподобань користувачів – враховуючи, що вподобання користувачів можуть змінюватися з часом, важливо враховувати неявні фідбеки, такі як останні дії та поведінка користувачів, для адаптації рекомендаційних систем до зміни інтересів користувачів.

3. Специфіка продукту або послуги: для деяких продуктів або послуг явні фідбеки можуть бути більш важливими, оскільки вони містять детальну інформацію про досвід користувачів. У таких випадках рекомендаційна система може надавати перевагу явним фідбекам. З іншого боку, для продуктів або послуг, які користувачі вживають швидко або часто (музика або новини), неявні фідбеки можуть бути кориснішими, оскільки вони дозволяють рекомендаційній системі оперативно реагувати на змінні інтереси користувачів. В таких випадках, система може надавати перевагу неявним фідбекам або комбінувати явні та неявні фідбеки для підтримки релевантності рекомендацій.

Для створення рекомендаційних систем можуть бути використані різні алгоритми, що враховують як явні, так і неявні вподобання користувачів. Шляхом комбінації різних підходів та врахування специфіки домену можна досягти більшої точності та персоналізації рекомендацій, що ефективність рекомендаційних систем.

У підсумку, вибір між явними та неявними видами фідбеків залежить від специфіки рекомендаційної системи, доступності даних та динаміки вподобань користувачів. Важливо розуміти, що ідеальне рішення часто полягає в поєднанні явних та неявних фідбеків [7], яке дозволяє підвищити точність, релевантність та адаптивність рекомендацій для кожного користувача.

## 18.4. Особливості надання оцінки значимості явним видам фідбеку

Явний фідбек від користувачів, такий як оцінки, коментарі та відгуки, є важливою інформацією для рекомендаційних систем. Однак, збір, аналіз та врахування цієї інформації може мати свої особливості.

Збір явних фідбеків може відбуватися через різні механізми, такі як форми оцінювання продуктів, написання відгуків, анкети чи опитування. Важливо спроектувати процес збору фідбеків таким чином, щоб забезпечити зручність для користувачів та стимулювати їх до надання відгуків.

Ключові аспекти значимості явних фідбеків:

1. Відстеження змін у вподобаннях користувачів – оцінки значимості явних фідбеків можуть допомогти відстежувати зміни вподобань користувачів протягом часу. Враховуючи динаміку оцінок або відгуків, рекомендаційні системи можуть підлаштовуватися до змінних інтересів користувачів, пропонуючи більш актуальні рекомендації.

2. Залучення користувачів – явні фідбеки можуть стимулювати користувачів до активної участі в процесі рекомендацій. Заохочуючи користувачів надавати свої відгуки та оцінки, система може отримати більше даних для аналізу та створення кращих рекомендацій.

3. Різноманітність рекомендацій – надання значимості явним видам фідбеку може забезпечити різноманітність рекомендацій, оскільки це допомагає системі враховувати різні аспекти вподобань користувачів. Таким чином, система може пропонувати не тільки популярні продукти або контент, а й менш відомі або навіть нішеві елементи, які можуть відповідати індивідуальним

інтересам користувачів. Частково це може допомогти вирішити проблему «довгого хвосту».

4. Відповідність контексту – за допомогою явних фідбеків рекомендаційні системи можуть краще враховувати контекст, в якому відбувається споживання продуктів або контенту. Наприклад, користувачі можуть надавати відгуки про фільми, які вони дивилися з друзями, або книги, які вони читали. Залежно від контексту, ця інформація може допомогти системі робити більш точні рекомендації.

5. Баланс між явними та неявними фідбеками – важливо враховувати, що явні фідбеки можуть бути менш доступними або менш точними за неявні. Тому, при врахуванні значимості явних фідбеків, слід також збалансувати їх з неявними фідбеками для отримання оптимальних результатів.

Надання значимості явним видам фідбеку в гібридних рекомендаційних системах може допомогти створювати більш точні, різноманітні та контекстно-залежні рекомендації. Однак, важливо знайти оптимальний баланс між явними та неявними фідбеками для досягнення найкращих результатів.

### 18.4.1. Збір оцінок та відгуків користувача

Збір явних фідбеків відіграє важливу роль в рекомендаційних системах, оскільки вони допомагають виявити інтереси користувачів та враховувати їх у рекомендаціях [8]. Нижче наведені кілька методів збору оцінок та відгуків користувачів для рекомендаційної системи:

1. Оцінки:

- Користувачі оцінюють фільми, товари тощо за шкалою (наприклад, від 1 до 10 зірок).
- Збереження історії оцінок користувачів для подальшого аналізу та використання у рекомендаційній системі.

2. Відгуки:

- Користувачі залишають текстові відгуки на товари чи контент.
- Застосовується аналіз сентиментів для визначення позитивного чи негативного характеру відгуків та врахування їх у рекомендаціях.

3. Анкети та опитування:

- Користувачі заповнюють анкети або опитування, які допоможуть зібрати інформацію про особисті смаки та інтереси користувачів.
- Аналіз відповідей користувачів, щоб краще зрозуміти їхні потреби та впроваджувати ці дані у рекомендаційну систему.

Збір явних фідбеків користувачів, таких як оцінки та відгуки, має свої особливості, які важливо враховувати при розробці та впровадженні рекомендаційних систем. Ось деякі з них:

1. Залучення користувачів – щоб збирати явні фідбеки, необхідно стимулювати користувачів активно висловлювати свою думку про продукти або контент. Це можна зробити, надаючи зручні інструменти для надання

оцінок та відгуків, а також заохочуючи їх за допомогою програм лояльності, бонусів або інших стимулів.

2. Упередженість та зміщення – при зборі явних фідбеків потрібно враховувати можливу упередженість та зміщення [9], такі як ефект самовідбору (користувачі, що оцінюють щось в системі, можуть мати інші вподобання, ніж загальна аудиторія) або ефект зграї (оцінки користувачів схильні наближатися до середнього значення через соціальний вплив).

3. Контроль якості – збір явних фідбеків вимагає контролю якості, щоб виявити і відфільтрувати спам та недостовірні дані. Використання модераторів, автоматичних фільтрів або комбінації цих методів може покращити достовірність та корисність зібраних даних.

4. Часові зміни – вподобання користувачів можуть змінюватися з часом, тому важливо забезпечити оновлення зібраних даних. Рекомендаційні системи повинні враховувати нові оцінки та відгуки, щоб адаптувати рекомендації до змін інтересів користувачів. Можна використовувати часові моделі, які забезпечують більшу вагу останнім оцінкам та відгукам, або періодично оновлювати модель рекомендаційної системи, щоб враховувати свіжі дані.

5. Проблема холодного старту – для нових користувачів або продуктів може бути замало доступних оцінок та відгуків для генерації рекомендацій. Це відомо як проблема холодного старту. У таких випадках для генерування початкових рекомендацій можна використовувати інші види інформації, такі як демографічні дані або описи продуктів.

6. Агрегування фідбеку – для підвищення точності рекомендаційних систем на основі явних фідбеків можна агрегувати декілька видів фідбеку. Наприклад, оцінки можуть бути поєднані з відгуками, таким чином, аналізуючи текст на настрій або ключові слова, можна отримати більш комплексне уявлення про вподобання користувачів.

Для вирішення проблем зі збором явних фідбеків користувача для рекомендаційних систем, можна використовувати наступні стратегії:

1. Спрощення процесу надання фідбеку – необхідно забезпечити простий та зручний інтерфейс для користувачів, щоб заохотити їх залишати оцінки та відгуки. Наприклад, використання зіркової системи оцінювання або кнопок «подобається» та «не подобається».

2. Інтеграція з соціальними мережами – залучення користувачів до надання фідбеку через соціальні мережі, якими вони вже користуються, або можливість користувачам швидко поширювати свої оцінки та відгуки.

3. Використання гейміфікації – застосування елементів гейміфікації, таких як нагород, балів або рейтингів, для заохочення користувачів надавати фідбек.

4. Забезпечення прозорості – необхідно пояснити користувачам, як їхній фідбек використовується для покращення рекомендацій, їм необхідно дати можливість налаштовувати свої вподобання.

5. Використання тимчасових рекомендацій - для нових користувачів або продуктів, які стикаються з проблемою холодного старту, можна

використовувати тимчасові рекомендації на основі демографічних даних або описів продуктів [10]. За мірою накопичення явних фідбеків можна перейти до рекомендацій на основі колаб оративної фільтрації або інших методів рекомендацій. Це допоможе покращити якість рекомендацій з часом, коли користувачі будуть залишати більше оцінок та відгуків.

6. Постійне оновлення моделі – для підтримки актуальності рекомендацій, треба періодично оновлювати модель рекомендаційної системи, інтегруючи нові оцінки та відгуки. Це допоможе системі реагувати на зміни вподобань користувачів та враховувати нові продукти або контент.

Застосування цих стратегій допоможе вирішити проблеми зі збором явних фідбеків користувача для рекомендаційних систем, покращуючи якість рекомендацій та задоволення користувачів.

### 18.4.2. Визначення ефективності та значимості оцінок та відгуків користувача

Ефективність явних фідбеків можна виміряти за допомогою таких метрик, як точність, повнота, F1-міра, RMSE та PCC, залежно від типу рекомендаційної системи та даних. Застосування цих метрик допомагає зрозуміти, наскільки відповідні рекомендації відображають інтереси користувачів на основі їх явних оцінок та відгуків.

Значимість явних фідбеків може бути виміряна за допомогою аналізу кореляції між рекомендаціями та оцінками, наданими користувачами. Це допомагає виявити, які оцінки та відгуки найбільш впливають на рекомендації та відповідають вподобанням користувачів.

Для підвищення ефективності рекомендаційних систем, можна провести відбір оцінок та відгуків на основі їх релевантності та значимості. Такий підхід допоможе сконцентруватися на найважливіших фідбеках користувачів, покращуючи якість рекомендацій та забезпечуючи більш відповідний досвід для користувачів. Відбір релевантних оцінок та відгуків може включати врахування таких факторів, як:

1. Часова актуальність – відгуки та оцінки, надані нещодавно, можуть бути більш важливими для рекомендацій, оскільки вони відображають актуальні інтереси користувачів.
2. Згідність думок – оцінки та відгуки, які суперечать більшості інших користувачів, можуть вказувати на менш значимі або непереконливі думки, тому їх вага може бути зменшена.
3. Авторитет користувача – оцінки та відгуки від користувачів з високим авторитетом або досвідом у відповідній галузі можуть надаватися більшої ваги, оскільки вони можуть бути більш об'єктивними та інформативними.

У відповідь на ці вимоги, можуть бути застосовані алгоритми для визначення ваги та релевантності явних фідбеків користувачів. Це допоможе рекомендаційним системам ефективніше використовувати оцінки та відгуки

користувачів для формування рекомендацій, які краще відповідають інтересам користувачів.

### 18.5. Особливості надання оцінки значимості неявним видам фідбеку

В контексті ефективності використання явних та неявних користувацьких фідбеків для гібридних рекомендаційних систем, оцінка значимості неявних видів фідбеку відіграє важливу роль. Неявні фідбеки відображають поведінку користувачів [11] під час взаємодії з продуктами або контентом, такими як перегляди сторінок, час перебування на сторінці, кількість натискань тощо. Оцінка значимості неявних фідбеків включає такі особливості:

1. Врахування контексту – неявні фідбеки можуть залежати від контексту, в якому користувач взаємодіє з продуктом або контентом. Наприклад, час дня, день тижня, або поточні події можуть впливати на інтереси користувачів.
2. Розрізнення різних типів взаємодії – різні види неявних фідбеків можуть мати різну значимість для рекомендаційної системи. Наприклад, додавання продукту до кошика може свідчити про більш сильний інтерес користувача, ніж просто перегляд сторінки продукту.
3. Нормалізація фідбеків – користувачі можуть мати різні моделі взаємодії, і це повинно бути враховано при оцінці значимості неявних фідбеків. Наприклад, деякі користувачі можуть часто натискати на рекомендовані продукти, тоді як інші можуть робити це менш активно. Для врахування цих індивідуальних відмінностей рекомендаційні системи можуть нормалізувати неявні фідбеки, порівнюючи поведінку користувача із загальними статистичними показниками або відносно інших користувачів.
4. Відслідковування змін у поведінці користувачів – значимість неявних фідбеків може змінюватися з часом в залежності від актуальних інтересів користувачів.
5. Застосування ваги до різних джерел фідбеку – у гібридних рекомендаційних системах, які використовують як явні, так і неявні фідбеки, може бути важливо надати різні ваги різним видам фідбеку.

Враховуючи вищенаведені особливості, розробники рекомендаційних систем можуть покращити ефективність використання неявних фідбеків та забезпечити кращі рекомендації користувачам.

#### 18.5.1. Збір даних про поведінку користувача та його взаємодії

Для збору даних про поведінку користувача рекомендаційні системи можуть використовувати різні методи, такі як:

1. Записи веб-журналів про дії користувачів на веб-сайті, такі як перегляди сторінок, натискання, додавання в кошик та покупки.
2. Системи можуть відстежувати дії користувачів в реальному часі, збираючи дані про взаємодії з продуктами, контентом або рекламою.

3. Аналіз активності користувачів у соціальних медіа, таких як відгуки, коментарі, лайки та репости, може допомогти системам зрозуміти інтереси та уподобання користувачів.

Після збору даних про поведінку користувача та його взаємодії, рекомендаційні системи повинні обробити та аналізувати ці дані. Зазвичай це може включати такі процеси, як:

1. Попередня обробка даних – очищення даних від шуму, видалення дублікатів, заповнення пропущених значень та нормалізація даних є важливими кроками попередньої обробки.

2. Виявлення шаблонів та аномалій – аналіз даних про поведінку користувача може виявити звичні шаблони поведінки, що свідчать про інтереси та уподобання користувачів. Також важливо виявляти аномальні дії, які можуть вказувати на спам, шахрайство або помилки в даних.

3. Кластеризація та сегментація користувачів – застосування методів кластеризації та сегментації може допомогти групувати користувачів за спільними інтересами, поведінкою або характеристиками.

Неявні фідбеки, зібрані з даних про поведінку користувача та його взаємодії, можуть бути інтегровані в гібридні рекомендаційні системи. Це може покращити рекомендаційну здатність системи, доповнюючи явні фідбеки користувачів [12]. Залежно від конкретної рекомендаційної системи, неявні фідбеки можуть бути використані для:

1. Зміцнення явних оцінок – інтеграція неявних фідбеків з явними оцінками може допомогти врахувати відмінності в поведінці користувачів та відображати їх справжні уподобання. Наприклад, якщо користувач часто переглядає певний тип продуктів, але не оцінює їх високо, система може скоригувати рекомендації, збільшивши вагу неявного фідбеку.

2. Визначення контексту – неявні фідбеки можуть допомогти системам враховувати контекст користувача, наприклад, час доби, день тижня, пристрій чи місцезнаходження. Це може поліпшити релевантність рекомендацій, забезпечуючи контент, який відповідає контексту користувача.

3. Виявлення нових інтересів – використання неявних фідбеків може допомогти виявити нові інтереси користувачів, які ще не відображені в їх явних оцінках. Системи можуть рекомендувати продукти або контент, які користувачі не розглядали раніше, але можуть стати цікавими на основі їх неявної поведінки.

4. Усунення упереджень – інтеграція неявних фідбеків може допомогти виявити та усунути упередження, що виникають через явні оцінки користувачів. Наприклад, користувачі можуть надавати вищі оцінки популярним продуктам або занижувати оцінки конкуруючим брендам.

### 18.5.2. Визначення ефективності та значимості неявних фідбеків користувача

Оцінка ефективності та значимості неявних фідбеків користувача має важливе значення для гібридних рекомендаційних систем, оскільки вони

допомагають зрозуміти, наскільки дійсно корисні та інформативні ці види фідбеків. У цьому розділі розглядаються ключові аспекти оцінки ефективності та значимості неявних фідбеків користувача.

1. Вимірювання корисності – оцінка корисності неявних фідбеків вимагає вивчення взаємодії користувачів з рекомендованим контентом. Метрики, які можуть використовуватися для цього, включають час перегляду, кількість кліків, частоту відвідування сторінки та інші. Застосування таких метрик дозволяє визначити ступінь зацікавленості користувачів у рекомендованому контенті та наскільки ефективно система адаптується до їхніх потреб.

2. Ранжування значимості – важливо визначити, які види неявних фідбеків найбільше впливають на рекомендації [13]. Це можна зробити за допомогою методів, таких як важливість ознак або аналіз кореляції між різними видами фідбеків та реакцією користувачів на рекомендації. Таким чином, можна виявити найбільш важливі види неявних фідбеків та зосередитися на їх використанні для підвищення ефективності рекомендацій.

3. Оцінка стабільності – неявні фідбеки можуть бути змінними та непостійними, що може впливати на стабільність рекомендацій. Наприклад, користувач може мати різні інтереси в різні часи дня або день тижня, або ж просто змінювати свої уподобання. Оцінка стабільності вимагає відстеження змін у неявних фідбеках користувачів протягом часу та аналізу впливу цих змін на якість рекомендацій.

4. Врахування контексту – важливо забезпечити, що значимість неявних фідбеків враховується в контексті особливостей користувача та рекомендованого контенту [14]. Наприклад, користувач може більше клікати на рекламний контент, але це не означає, що реклама є найкращим варіантом для рекомендацій. Застосування контекстно-залежних моделей, які враховують такі фактори, може допомогти забезпечити більш точні та ефективні рекомендації.

5. Валідація результатів – для переконання у ефективності та значимості неявних фідбеків користувача, важливо проводити регулярну валідацію результатів. Це може включати порівняння рекомендацій, отриманих на основі явних та неявних фідбеків, та оцінку їх збігу, а також використання A/B-тестування для вимірювання впливу різних комбінацій фідбеків на відгук користувачів.

Визначення ефективності та значимості неявних фідбеків користувача вимагає комплексного підходу, що може включати відстеження корисності рекомендацій, ранжування значимості різних видів фідбеків, оцінку стабільності та контексту, а також регулярну валідацію результатів.

Під час розробки гібридних рекомендаційних систем, інтеграція та аналіз явних та неявних користувацьких фідбеків може значно підвищити ефективність та точність рекомендацій. Врахування особливостей обох видів фідбеків дозволяє створити рішення, які надають більш персоналізовані та адаптивні рекомендації, що відповідають різним потребам та уподобанням користувачів.

## 18.6. Висновки

В результаті аналізу ефективності використання явних та неявних користувацьких фідбеків для гібридних рекомендаційних систем можна відзначити наступні ключові моменти:

1. Явні та неявні фідбеки мають свої переваги та недоліки, і їх використання залежить від специфіки продукту чи контенту та доступності даних. Явні фідбеки, такі як оцінки та відгуки, можуть бути більш точними та зрозумілими, проте можуть також страждати від упереджень користувачів та низької активності. Неявні фідбеки, як-от перегляди та кліки, можуть відображати реальну поведінку користувачів, але можуть бути менш стабільними та змінними.
2. Гібридні рекомендаційні системи, які інтегрують явні та неявні фідбеки, можуть підвищити точність та різноманітність рекомендацій, адаптуватися до змін уподобань користувачів та забезпечити більш персоналізовані рекомендації.
3. Визначення ефективності та значимості явних та неявних фідбеків вимагає комплексного підходу, що включає відстеження корисності рекомендацій, ранжування значимості різних видів фідбеків, оцінку стабільності та контексту, а також регулярну валідацію результатів.
4. Збір даних про поведінку користувача та його взаємодії з продуктами або контентом, а також їх адекватна обробка, є важливими етапами створення ефективних рекомендаційних систем.
5. Врахування обох видів фідбеків може допомогти рекомендаційним системам уникнути недоліків, пов'язаних лише з одним видом відгуків. Наприклад, занадто велика залежність від явних оцінок може призвести до забарвлених або обмежених рекомендацій, тоді як переважний акцент на неявних фідбеках може створити шум та нестабільність у системі. Комбінація обох видів фідбеків дозволяє створити більш збалансовану, стабільну та точну рекомендаційну систему.

У цілому, гібридні рекомендаційні системи, які використовують явні та неявні користувацькі фідбеки, виявилися ефективним підходом для підвищення точності, різноманітності та адаптивності рекомендацій. Це дає можливість створити більш персоналізовані рекомендації, які відповідають змінам уподобань користувачів і покращують загальний досвід користувачів. Але розробники повинні враховувати специфіку продукту чи контенту, доступність даних та цілі системи, щоб забезпечити максимально ефективні рекомендації для користувачів.

## Література

1. Lendave, V. (2020). How To Measure The Success Of A Recommendation System? [Online]. Available at: https://analyticsindiamag.com/how-to-measure-the-success-of-a-recommendation-system/.

2. Abdollahpouri, H., Burke, R., & Mobasher, B. (2020). Multistakeholder recommendation: Survey and research directions, pp. 12-13. [Online]. Available at: https://link.springer.com/article/10.1007/s11257-019-09256-1.

3. Park, J. H., & Choi, M. J. (2017). Analysis of implicit feedbacks for music recommendation. Multimedia Tools and Applications, 76(24), 26291-26300. [Online] Available at: https://www.researchgate.net/publication/341702953_Comparison_of_implicit_and_explicit_feedback_from_an_online_music_recommendation_service.

4. Kumar, S. (2022). Explicit vs Implicit Collaborative Filtering: Explained. Sumit Kumar's Blog. [Online]. Available at: https://blog.reachsumit.com/posts/2022/09/explicit-implicit-cf/.

5. Z. Yusefi Hafshejani, M. Kaedi, and A. Fatemi, "Improving sparsity and new user problems in collaborative filtering by clustering the personality factors," Electronic Commerce Research, vol. 18, no. 4, pp. 813–836, dec 2018. [Online]. Available at: http://link.springer.com/10.1007/s10660-018-9287-x.

6. M. J. Shayegan and M. Valizadeh, "A Recommender System based on the analysis of personality traits in Telegram social network," oct 2020. [Online]. Available at: http://arxiv.org/abs/2010.00643.

7. Guo, G., Zhang, J., & Thalmann, D. (2017). Merging trust in collaborative filtering to alleviate data sparsity and cold start. Knowledge-Based Systems, 115, 57-68. [Online]. Available at: https://www.sciencedirect.com/science/article/abs/pii/S0950705113003870.

8. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web (pp. 173-182). [Online] Available at: https://dl.acm.org/doi/10.1145/3038912.3052569.

9. Kang, W. C., & McAuley, J. (2018). Self-attentive sequential recommendation. In 2018 IEEE International Conference on Data Mining (ICDM) (pp. 197-206). IEEE. [Online]. Available at: https://ieeexplore.ieee.org/document/8594844.

10. Beutel, A., Covington, P., Jain, S., Xu, C., Li, J., Gatto, V., & Chi, E. H. (2018). Latent cross: Making use of context in recurrent recommender systems. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (pp. 42-54). [Online]. Available at: https://dl.acm.org/doi/10.1145/3159652.3159727.

11. Nguyen, P., Tomeo, P., Noia, T. D., & Sciascio, E. D. (2017). Content-based recommendations via DBpedia and Freebase: A case study in the music domain. Multimedia Tools and Applications, 76(4), 5487-5522. Available at: https://sisinflab.poliba.it/publications/2015/NTDD15/Content-based%20recommendations%20via%20DBpedia%20and%20Freebase%20a%20case%20study%20in%20the%20music%20domain.pdf.

12. H. Zhu, L. Li, H. Jiang, and A. Tan, "Inferring Personality Traits from Attentive Regions of User Liked Images Via Weakly Supervised Dual Convolutional Network," Neural Processing Letters, vol. 51, no. 3, pp. 2105–2121, jun 2020. [Online]. Available at: http://link.springer.com/10.1007/s11063-019-09987-7.

13. M. Braunhofer, M. Elahi, and F. Ricci, “Usability Assessment of a Context-Aware and Personality-Based Mobile Recommender System,” in E-Commerce and Web Technologies, M. Hepp and Y. Hoffner, Eds. Cham: Springer International Publishing, 2014, pp. 77–88. [Online]. Available at: https://link.springer.com/chapter/10.1007/978-3-319-10491-1_9.

14. H. Feng and X. Qian, “Recommendation via user’s personality and social contextual,” in Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM ’13. New York, New York, USA: ACM Press, 2013, pp. 1521–1524. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2505515.2507834.

А. І. Абакумов
Є. В. Бабешко
Ніколаос Бардіс
І. О.Васильєв
О. О. Вдовіченко
О. Ю. Веприцька
В. В. Гаєвський
К. С. Гайдук
О. О. Гордєєв
С. І. Доценко
Г. А. Землянко
Марк Ізраель
О. О. Ілляшенко
Є. О. Канарський
І. М. Клюшніков
А. І. Кулягін
Т. В. Кунуп
Г. А. Кучук
Є. В. Мерзлікін
О. І. Морозова
В. В. Нарожний
О. С. Неретін
О. О. Орєхов
А. Є. Перепелицин
В. Я. Пєвнєв
Д. О. Сверчков
С. В. Скоробогатько
А. О. Стадник
Г. Л. Федоренко
Г. В. Фесенко
В. С. Харченко
О. А. Чуйко
В. Р. Щеглов

# МЕТОДИ ТА ТЕХНОЛОГІЇ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ТА БЕЗПЕКИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

Монографія
(українською мовою)

**Редактор В. С. Харченко, О. І. Морозова**

Комп'ютерна верстка
О. І. Морозова

Зв. план, 2023
Підписаний до друку 03.07.2023 Формат 60x84 1/16.
Папір офс. №2. Офс. друк. Гарнітура Times New Roman.
Умов. друк. арк. 20,46. Обл.-вид. арк. 22,0.
Наклад 100 прим. Замовлення № 03072023.

---

Зверстано і надруковано в ТОВ "Видавництво "Юстон"
01034, м. Київ, пр. Перемоги, 62-Б, оф. 2 тел.: (044) 360-22-66,
www.yuston.com.ua

Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру видавців,
виготовлювачів і розповсюджувачів видавничої продукції
серія ДК № 4973 від 09.09.2015 р.