

Вінницький національний технічний університет  
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації  
(повне найменування інституту, назва факультету (відділення))

Кафедра автоматизації та інтелектуальних інформаційних технологій  
(повна назва кафедри (предметної, циклової комісії))

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**  
на тему:

**«Розробка і дослідження інтелектуальних методів опису інформації в комунікаційних мережах автоматизованих систем»**

Виконав: магістр 2 курсу, групи ЗАКІТ-22м  
спеціальності 151- Автоматизація та  
комп'ютерно- інтегровані технології»  
(шифр і назва напрямку підготовки, спеціальності)

Андрій ЗАПЛЕТНЮК  
(ім'я ПРІЗВИЩЕ)

Керівник: к.т.н., професор каф. КСУ

Микола БИКОВ  
(ім'я ПРІЗВИЩЕ)

«14» грудня 2023 р.

Опонент: к.т.н., доцент каф. КН

Володимир ОЗЕРАНСЬКИЙ  
(ім'я ПРІЗВИЩЕ)

«14» грудня 2023 р.

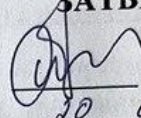
Допущено до захисту  
Завідувач кафедри АІТ  
Олег БІСІКАЛО  
(ім'я ПРІЗВИЩЕ)  
«18» грудня 2023 року

Вінниця ВНТУ – 2023 рік



Вінницький національний технічний університет  
 Факультет інтелектуальних інформаційних технологій та автоматизації  
 Кафедра автоматизації та інтелектуальних інформаційних технологій  
 Рівень вищої освіти II-й (магістерський)  
 Галузь знань – 15 Автоматика та приладобудування  
 Спеціальність - 151 Автоматизація та комп'ютерно-інтегровані технології  
 Освітня програма - Інформаційні системи і інтернет речей

**ЗАТВЕРДЖУЮ**

 Зав. кафедри АІТ  
 проф. Олег БІСКАЛО  
 20 09 2023 р.

### **ЗАВДАННЯ** НА МАГІСТЕРСКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Заплетнюку Андрію Вікторівичу

(прізвище, ім'я, по батькові)

1. Тема роботи «Розробка і дослідження інтелектуальних методів опису інформації в комунікаційних мережах автоматизованих систем»

керівник роботи Биков Микола Максимович, к. т. н., професор  
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)  
 затверджені наказом ВНТУ від "18" вересня 2023 року № 247

2. Строк подання студентом роботи 12.12. 2023 року

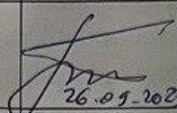
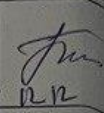
3. Вихідні дані до роботи: стани об'єктів управління представлені ранговими конфігураціями; досліджувані моделі опису інформації – моделі рангових конфігурацій і кодів; середовища моделювання – Visual C++, PyCharm, Microsoft Visual Studio 2017. (числові дані)

4. Зміст текстової частини: Анотація; Вступ; Техніко-економічне обґрунтування вибраного опису інформації в системах прийняття рішень; Математичне обґрунтування вибраного методу представлення інформації; Розробка алгоритмічного та програмного забезпечення; Експериментальна частина.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень) 1. Математичні моделі рангових конфігурацій і рангових кодів–1 плакат. 2. Приклади рангових кодів і рангових конфігурацій–1 плакат; 3. Схеми програм дослідження інтервальної моделі–6 аркушів; 4. Схема пристрою передачі рангової інформації – 1 аркуш; 5. Схема декодера – 1 аркуш. 6. Схеми програм кодування і декодування рангових конфігурацій – 2 аркуші; 7. Машинний експеримент – 2 плакати.



## 6. Консультанти розділів магістерської кваліфікаційної роботи

Розділ змістової частини роботи	Ім'я, прізвище та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-3	Микола БИКОВ к.т.н., професор каф. КСУ	 26.09.2023	 26.09.2023

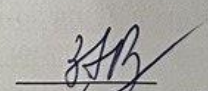
7. Дата видачі завдання " 26 " 09 2023 року

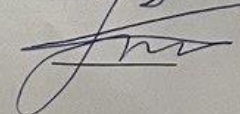
## КАЛЕНДАРНИЙ ПЛАН

- |    |                                                                      |                |
|----|----------------------------------------------------------------------|----------------|
| 1  | Техніко-економічне обґрунтування доцільності розробки                | 02.10.2023 р.  |
| 2  | Постановка задачі і розробка технічного завдання                     | 04.10. 2023 р. |
| 3  | Математичне обґрунтування вибраного методу представлення інформації  | 20.10. 2023 р. |
| 4  | Розробка алгоритмічного та програмного забезпечення системи          | 15.11. 2023 р. |
| 5  | Машинний експеримент                                                 | 27.11. 2023 р. |
| 6  | Оформлення пояснювальної записки, графічного матеріалу і презентації | 30.11. 2023 р. |
| 7  | Апробація результатів дослідження                                    | 07.12.2023 р.  |
| 8  | Публікації                                                           | 10.12.2023 р.  |
| 9  | Графічні матеріали:                                                  |                |
|    | Плакати демонстраційні                                               | 01.12. 2023 р. |
|    | Схеми програм дослідження інтервальної моделі                        | 01.12. 2023 р. |
|    | Схема пристрою передачі інформації                                   | 02.12. 2023 р. |
|    | Схема декодера                                                       | 02.12. 2023 р. |
|    | Схема програми кодування рангової інформації                         | 04.12. 2023 р. |
|    | Схема програми декодування рангової інформації                       | 05.12. 2023 р. |
|    | Результати машиноого експерименту                                    | 06.12. 2023 р. |
| 10 | Захист МКР                                                           | 12.12. 2023 р. |

Студент

Керівник роботи


 Андрій ЗАПЛЕТНЮК


 Микола БИКОВ

## АНОТАЦІЯ

УДК 004.93

Заплетнюк А. В. Розробка і дослідження інтелектуальних методів опису інформації в комунікаційних мережах автоматизованих систем. Магістерська кваліфікаційна робота зі спеціальності 151 – автоматизація та комп'ютерно-інтегровані технології, освітня програма – інтелектуальні комп'ютерні системи. Вінниця: ВНТУ, 2023. 102 с.

На укр. мові. Бібліогр.: 33 назв; рис.: 23.

Магістерська кваліфікаційна робота присвячена розробці ефективних методів, моделей і засобів опису і передавання інформації в системах автоматизованого управління.

Виконано математичне обґрунтування запропонованих методів і моделей рангового опису і кодування інформації, проведено дослідження їх адекватності

Проведено дослідження запропонованих методів і моделей, розроблено пристрій передачі інформації з використанням запропонованих кодів, розроблено схему декодера рангового коду.

У розділі програмного забезпечення на базі проведених експериментальних і теоретичних досліджень характеристик запропонованих кодів розроблено алгоритми кодування і декодування інформації, представленої ранговим кодом. Також розроблено алгоритм локалізації і виправлення помилок в кодах, внесених каналом передачі. Розроблені алгоритми реалізовані програмами, написаними мовою Сі.

Ілюстративна частина містить 10 плакатів і креслень з ілюстрацією результатів роботи.

Ключові слова: методи опису інформації, комунікаційні мережі, моделювання, рангові конфігурації, DRP-код, кодування, нейронні мережі, декодування, виявлення і корекція помилок, автоматизовані системи.

## ABSTRACT

Zapletniuk A. V. Development and research of intelligent methods of information description in communication networks of automated systems. Master's thesis on specialty 151 - automation and computer-integrated technologies, educational program - intelligent computer systems. Vinnytsia: VNTU, 2023. 102 p.

In Ukrainian speech Bibliography: 33 titles; Fig.: 23.

The master's thesis is devoted to the development of effective methods, models and means of description and transmission of information in automated control systems.

Mathematical substantiation of the proposed methods and models of rank description and coding of information was carried out, and a study of their adequacy was carried out

A study of the proposed methods and models was carried out, an information transmission device using the proposed codes was developed, and a ranking code decoder scheme was developed.

In the software section, on the basis of experimental and theoretical studies of the characteristics of the proposed codes, algorithms for encoding and decoding the information represented by the rank code have been developed. An algorithm for localization and correction of errors in the codes introduced by the transmission channel has also been developed. The developed algorithms are implemented by programs written in the C language.

The illustrative part contains 10 posters and drawings illustrating the results of the work.

Keywords: information description methods, communication networks, modeling, rank configurations, DRP code, coding, neural networks, decoding, error detection and correction, automated systems.

## ЗМІСТ

ВСТУП.....	8
<b>1 ВАРІАНТНИЙ АНАЛІЗ МЕТОДІВ ОПИСУ ІНФОРМАЦІЇ</b>	
<b>В КОМУНІКАЦІЙНИХ МЕРЕЖАХ АВТОМАТИЗОВАНИХ СИСТЕМ....</b>	
1.1 Суть технічної проблеми, що виникла на сучасному етапі розвитку науки .....	12
1.2 Аналіз методів кодування інформації в комунікаційних мережах автоматизованих систем управління.....	14
1.3 Підвищення завадостійкості і інформаційної ємності каналів зв'язку комунікаційних мереж запропонованим методом кодування.	21
1.4 Підвищення ефективності прийняття управлінських рішень за рахунок запропонованого методу опису і кодування даних.....	24
1.5 Висновки до розділу.....	31
<b>2 РОЗРОБКА І МАТЕМАТИЧНЕ ОБГРУНТУВАННЯ ЗАПРОПОНОВАНИХ МЕТОДІВ ТА ЗАСОБІВ ОПИСУ І ПЕРЕДАЧІ ІНФОРМАЦІЇ.....</b>	
2.1 Математичне обґрунтування рангового опису інформації.....	33
2.2 Розробка структурної схеми пристрою передавання інформації....	42
2.3 Опис моделі кодера рангової інформації.....	45
2.4 Використання нейронної мережі в якості кодера.....	56
2.5 Дослідження коригуючих властивостей рангового коду.....	60
2.6 Висновки до розділу.....	67
<b>3 РОЗРОБКА АЛГОРИТМІВ І ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....</b>	
3.1 Розробка алгоритму кодування рангових конфігурацій <i>DRP</i> -кодом.....	69
3.2 Розробка алгоритму виправлення помилок в <i>DRP</i> -кодi.....	72
3.3 Розробка алгоритму декодування <i>DRP</i> -коду.....	73
3.4 Розробка програмного забезпечення.....	75
3.5 Результати тестування програми.....	76

	7
3.6 Інструкція користувачу.....	76
ВИСНОВКИ.....	78
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	80
ДОДАТКИ.....	83
ДОДАТОК А (обов'язковий) Технічне завдання.....	84
ДОДАТОК Б (обов'язковий) Ілюстративна частина.....	87
ДОДАТОК В (обов'язковий) Лістинг програми.....	98
ДОДАТОК Г (обов'язковий) Протокол перевірки МКР .....	106

## ВСТУП

**Актуальність теми.** Сучасний етап розвитку інтелектуальних інформаційних технологій і інформаційних систем, призначених задовольнити потреби суспільства у фіксації, зберіганні, передаванні та обробці великих масивів даних, необхідних для прийняття управлінських рішень, виникла необхідність вдосконалення функціонування системи «Інтернет речей». Так як об'єкти, інтегровані в Інтернет речей з допомогою ідентифікаторів, сенсорів та мереж в результаті еволюціонування породжують різні описи інформації, то це вимагає в кожному випадку розробки різних алгоритмів обробки даних для прийняття рішень за допомогою інтелектуальних комп'ютерних систем. Тому постає нагальна задача розробки моделей та методів для уніфікованого подання інформації, які можна було б використати для створення уніфікованих методів і алгоритмів аналізу даних, добування знань та їх ефективного використання для оперативного ухвалення управлінських рішень. Зазвичай управлінські рішення базуються на основі обробки інформації про відстань між поточним і бажаним станами об'єкта «Інтернету речей». Так як відстань в різних параметричних просторах (таких як імовірнісний, нечіткий, наближений, детерміністичний тощо) вимірюється по-різному, для кожного конкретного випадку потрібно розробляти свої методи та алгоритми для подання інформації та прийняття рішень. З урахуванням результатів, наведених у роботах [1-6], у даній роботі зроблено висновок, що для процесу ідентифікації станів об'єктів управління та оптимізації їх функціонування важливими є не просто відстані між станами, а їх ранговий порядок.

Для опису взаємозв'язку рангів відстаней між станами об'єктів було введено термін "рангова конфігурація". Для підвищення ефективності використання рангових конфігурацій у комп'ютерному аналізі даних та у прийнятті рішень, був використаний обчислювально ефективний двійковий метод представлення конфігурацій у вигляді двійкових рангових кодів.



На відміну стандартних кодів, які будуються з точки зору задоволення вимог розрізнюваності і завадостійкості, дані коди володіють деякими характеристиками інтелектуальності. Ці коди, подібно нейронам, які формують ваги відстаней один до одного, мають можливість запам'ятовувати інформацію про відстані між елементами, що підлягають кодуванню. Вони містять інформацію про самі стани об'єктів і також про відстані між ними в формі рангів. При впровадженні такого підходу виникає низка завдань, пов'язаних з необхідністю вивчення раніше запропонованих методів і моделей для уніфікованого опису інформації, а також визначення властивостей рангових кодів під час їх передачі через інформаційну мережу "Інтернет речей". Наведені вище фактори роблять тему даної магістерської кваліфікаційної роботи актуальною.

**Зв'язок роботи з науковими програмами, планами, темами.** Дослідження по даній темі виконані у відповідності з планом науково-дослідної роботи кафедри комп'ютерних систем управління.

**Мета магістерської роботи** полягає у поліпшенні оперативності ухвалення рішень у системах "Інтернет речей" шляхом розробки ефективних методів, моделей та засобів опису інформації та її достовірної передачі у комунікаційних мережах цих систем.

Для вирішення наступних задач потрібно досягнути поставленні цілі:

- провести аналіз методів опису даних в сучасних інформаційних системах;
- розробити математичне обґрунтування методів і моделей уніфікованого представлення інформації про стани об'єктів управління в різних параметричних просторах;
- провести аналіз для визначення того, наскільки запропоновані моделі уніфікованого опису інформації можуть охопити всі можливі конфігурації станів об'єктів;
- створити математичну модель нейронної мережі, яка буде виступати як кодер для опису інформації про стани систем у параметричному

просторі, перетворюючи їх у рангову конфігурацію, та провести її дослідження;

- розробити алгоритм кодування рангових конфігурацій *DRP*-кодом та здійснити теоретичну оцінку стійкості до перешкод та коригуючої здатності розроблених кодів.

**Об'єктом дослідження** є процеси опису інформації про стани об'єктів управління ранговими конфігураціями у вигляді рангових кодів

**Предметом дослідження** є методи, моделі, засоби та алгоритми кодування та передачі інформації в комунікаційних мережах автоматизованих систем.

**Методи дослідження.** Під час виконання досліджень використовувалися такі методи: дискретний аналіз; теорія множин; теорія інформації та теорія кодування; теорія ймовірностей; комп'ютерне моделювання та теорія алгоритмів.

**Новизна результатів досліджень.**

- запропоновано модифікований алгоритм опису інформації про стани об'єктів управління у вигляді рангових конфігурацій, що дозволяє однотипно реалізувати процес обробки інформації для даних, описаних в різних параметричних просторах;

- вперше виконано оцінку адекватності прийняття рішень на основі запропонованого опису інформації з використанням нейронної мережі;

- проведено теоретичну оцінку коригувальних можливостей запропонованого рангового коду з використанням відстані Хемінга;

- запропоновано метод локалізації і виправлення помилок, внесених комунікаційним каналом в повідомленні, представленим ранговим кодом.

**Практична цінність.** Розроблені в результаті проведених в роботі досліджень методи, моделі і апаратно-програмне забезпечення дозволяють підвищити оперативність прийняття рішень та підвищити завадостійкість і інформаційну ємність каналів передачі інформації в комунікаційних мережах систем автоматизації.

**Апробація результатів роботи.** Результати досліджень і отриманих положень доповідалися на Міжнародній науково-практичній Інтернет конференції студентів, аспірантів та молодих науковців ВНТУ «Молодь в науці» (2023). <https://conferences.vntu.edu.ua/index.php/mn/mn2024>



# 1 ВАРІАНТНИЙ АНАЛІЗ МЕТОДІВ ОПИСУ ІНФОРМЦІЇ В КОМУНІКАЦІЙНИХ МЕРЕЖАХ АВТОМАТИЗОВАНИХ СИСТЕМ

## 1.1 Суть технічної проблеми, що виникла на сучасному етапі розвитку науки

Розвиток систем контролю та управління, розвиток Інтернету речей з різними типами об'єктів призвели до постійного включення підходів до аналізу даних, їх представлення, обробки та прийняття рішень. Це пояснюється тим, що ці системи мають різні контрольні цілі та цілі управління, що ускладнює розробку ефективних методів і алгоритмів аналізу даних, вилучення знань і їх використання в процесі прийняття управлінських рішень, що часто викликає великі труднощі. У науково-технічній літературі доведено, що визначення стану об'єкта контролю та управління та оптимізацію його роботи можна здійснювати на основі інформації про відстань між поточним станом об'єкта та цільовим станом, стан у параметричному просторі. Доведення цього факту для просторів з детерміністичним і імовірнісним описами можна знайти, наприклад, в роботах авторів Штейнберга С. і Ту Дж. та Гонсалеса Р. [7,8], а для просторів з нечіткими і наближеними описами – в роботах авторів Коско Б., Краскела Дж., Павляка З., Дюбуа Д. [9,10,11,12]. Тому зазвичай рішення на керування в системах управління приймають на основі цієї інформації, застосовуючи оптимізаційні алгоритми переведення об'єкта керування з поточного стану в цільовий, наприклад, алгоритм динамічного програмування. Оскільки ці відстані в різних параметричних просторах (детерміністичному, імовірнісному, наближеному, нечіткому та ін.) описують по-різному, то це вимагає для кожного випадку розробки окремих методів і алгоритмів представлення інформації і прийняття рішень.

Дослідження [2,3,4] показали, що важливою інформацією в системах керування для реалізації процесів ідентифікації стану та прийняття рішень є

не відстань між цими станами, а відношення рангів між ними. Поняття рангової організації було введено для опису порядку відстаней між станами системи. Ефективність використання цих конструкцій для прийняття рішень і комп'ютерного аналізу даних залежить від того, як вони описані. Для цього було розроблено обчислювально ефективний метод двійкового кодування, який визначає об'єкти (стати) та містить інформацію про ранги відстаней між ними. Цей код називається потенційним або кодом DRP (код збереження рангу відстані). Термін «потенціал» використовувався через його подібність до електричного поля, в якому величина електричної взаємодії між зарядами залежить від величини потенціалу.

Запропонований у роботі код відрізняється від стандартного коду тим, що він має декілька інтелектуальних характеристик. Подібно до нейронів, які формують один з одним вагові відстані в нейронній мережі, ці коди мають здатність запам'ятовувати інформацію про відстані між елементами, що кодуються.

На основі вирішення реальних задач (ідентифікація станів системи, розпізнавання образів) використання запропонованого коду дозволяє створювати системи керування з різними способами опису вхідних даних (можливим, детермінованим, нечітким). Доведено, що можуть бути інтегровані алгоритми прийняття рішень - імовірнісні, апроксимовані ін.

Проведений аналіз, спрямований на математичну демонстрацію та розв'язання запропонованого методу для вирішення актуальної проблеми уніфікації методів представлення та алгоритмів обробки інформації в системах прийняття рішень з різними параметричними описами об'єктів керування, свідчить про необхідність додаткових досліджень. Є багато проблем із реальним впровадженням. Ці завдання повинні включати аналіз способів опису даних у сучасних автоматизованих системах та їх передачі в інформаційно-комунікаційних мережах систем обробки даних. Математично продемонструйте метод і модель для однакового опису інформації про різні параметри просторах ранговими конфігураціями; провести аналіз повноти

рангових кодів для покриття всієї множини можливих конфігурацій станів об'єкта управління; дослідити адекватність прийняття рішень на управління шляхом застосування запропонованого уніфікованого опису для кодування нейронів нейронної мережі; розробити алгоритми кодування рангової інформації потенціальним кодом і виконати теоретичну оцінку придатності розроблених кодів до виявлення і коригування помилок внесених комунікаційними каналами під час передачі.

## **1.2 Аналіз методів кодування інформації в комунікаційних мережах автоматизованих систем управління**

Зазвичай в комунікаційних мережах систем автоматизації використовують методи кодування, орієнтовані на забезпечення завадостійкості передачі інформації в фізичних каналах зв'язку, або для вирішення питань синхронізації передачі і прийому даних між передавальною і приймальнимю станціями мережі.

### **1.2.1 Аналіз принципів опису інформації завадостійкими кодами**

На разі є відомою велика кількість завадостійких кодів, які класифікуються за різними ознаками. Перешкодостійкі коди можна розділити на два великих класи: блокові та безперервні. При блочному кодуванні послідовність елементарних повідомлень джерела розбивається на відрізки і кожному відрізку ставиться у відповідність певна послідовність (блок) кодових символів, яка називається кодовою комбінацією. Множина кодових комбінацій, можливих при даному способі блокового кодування, утворює блоковий код.

Довжина блоку може бути як постійною, так і змінною. Розрізняють рівномірні та нерівномірні блокові коди. Перешкодостійкі коди є, як правило, рівномірними.



Блокові коди бувають роздільними та нероздільними. До роздільних відносяться коди, в яких символи за їх призначенням можуть бути поділені на інформаційні символи, що несуть інформацію про повідомлення та перевірочні. Такі коди позначаються як  $(n, k)$ , де  $n$ -довжина коду,  $k$ -число інформаційних символів. Можлива кількість комбінацій у кодї не перевищує значення  $2^k$ . Нероздільними вважаються коди, символи яких неможливо розглядати окремо за їхнім призначенням як інформаційні та перевірочні.

Коди з постійною вагою (КПВ) характеризуються тим, що їх кодові комбінації містять однакову кількість одиниць: Прикладом такого коду є код “3 із 7”, у якому кожна кодова комбінація містить три одиниці та чотири нулі (стандартний телеграфний код МТА- 3).

Коди з постійною вагою дозволяють виявити всі помилки кратності  $q=1, \dots, n$  за винятком випадків, коли кількість одиниць, що перейшли в нулі, дорівнює числу нулів, що перейшли в одиниці. У повністю асиметричних каналах, у яких має місце лише один вид помилок (перетворення нулів на одиниці чи одиниць на нулі), такий код дозволяє виявити всі помилки. У симетричних каналах ймовірність невиявленої помилки можна визначити як ймовірність одночасного спотворення однієї одиниці та одного нуля:

$$P_{nn} = C_n^2 P_n^2 (1 - P_n)^{n-2}$$

де  $P_{nn}$  - ймовірність невиявленої помилки.

Серед роздільних кодів розрізняють лінійні та нелінійні. До лінійних відносяться коди, в яких порозрядна сума за модулем 2 будь-яких двох кодових слів є кодовим словом. Лінійний код називається систематичним, якщо перші  $k$  символів його будь-якої кодової комбінації є інформаційними, інші  $(n-k)$  символів — перевірочними.

Серед лінійних систематичних кодів найбільш простий код  $(n, n-k)$ , що містить один перевірочний символ, який дорівнює сумі за модулем 2 всіх інформаційних символів. Цей код, який називається кодом з перевіркою на

парність, дозволяє виявити всі поєднання помилок непарної кратності. Імовірність невиявленої помилки у першому наближенні можна визначити як ймовірність спотворення двох символів:

Підкласом лінійних кодів є циклічні коди. Вони характеризуються тим, що всі набори, утворені циклічною перестановкою будь-якої кодової комбінації, є кодовими комбінаціями. Ця властивість дозволяє значною мірою спростити кодуєчий і декодуєчий пристрій, особливо при виявленні помилок і виправленні одиночної помилки. Прикладами циклічних кодів є коди Хеммінга, коди Боуза – Чоудхурі – Хоквінгема (БЧХ – коди) тощо.

Прикладом нелінійного коду є код Бергера, у якого перевірочні символи представляють двійковий запис числа одиниць у послідовності інформаційних символів. Наприклад, таким є код: 00000; 00101; 01001; 01110; 10001; 10110; 11010; 11111. Коди Бергера застосовуються в асиметричних каналах. У симетричних каналах вони виявляють усі поодинокі помилки та деяку частину багатократних.

Безперервні коди характеризуються тим, що операції кодування та декодування виробляються над безперервною послідовністю символів без розбиття її на блоки. Серед безперервних найбільш застосовні згорткові коди.

Як відомо розрізняють канали з незалежними і помилками, що групуються. Відповідно перешкодостійкі коди можна розбити на два класи: незалежні помилки, що виправляють, і пакети помилок, що виправляють. Далі будуть розглядатися переважно коди, що виправляють незалежні помилки. Це пояснюється тим, що хоча для виправлення пакетів помилок розроблено багато ефективних кодів, на практиці доцільніше використовувати коди, що виправляють незалежні помилки разом із пристроєм перемежування символів або декореляції помилок. При цьому символи кодової комбінації не передаються один за одним, перемішуються із символами інших кодових комбінацій. Якщо інтервал між символами, що належать одній кодовій комбінації, зробити більше ніж “пам'ять” каналу, то

помилки в межах кодової комбінації можна вважати незалежними, що дозволяє використовувати коди, що виправляють незалежні помилки.

Описані типи кодів проектувались з точки зору відповідності двом вимогам:

- 1- забезпечення розрізнюваності кодованих елементів;
- 2- забезпечення можливості виявлення і корекції помилок, внесених каналом зв'язку під час передачі інформації.

Тому такий опис інформації про стани системи автоматизованого управління не має ознак інтелектуальності, оскільки коди не пам'ятають співвідношення відстаней між станами системи, що не дає можливості приймати рішення на керування системою.

### **1.2.2 Аналіз характеристик методів кодування в комп'ютерних комунікаційних мережах систем автоматизації**

Для кодування інформації під час її передачі по комп'ютерних мережах систем автоматизації застосовують потенційні й імпульсні коди.

Потенційні коди використовують лише значення потенціалу сигналу для представлення логічних одиниць і нулів і не враховують їх відмінності для формування повного імпульсу. Імпульсні коди дозволяють представляти двійкові дані у вигляді імпульсів або частин імпульсів (різниці потенціалів у певному напрямку) певної полярності.

До методів кодування в комп'ютерних комунікаційних мережах висувають цілий ряд вимог, зумовлених необхідністю безпомилкової передачі інформації про стан об'єктів управління і синхронізації роботи передавального і приймального комп'ютерів.

При передачі дискретної інформації за допомогою прямокутних імпульсів необхідно вибрати метод кодування, який одночасно досягає кількох цілей:

- мав мінімальну постійну складову;
- сигнал мав найменшу доступну спектральну ширину за тієї самої



швидкості передачі даних.

- з'явилася функція виявлення помилок.
- забезпечує синхронізацію між передавачем і приймачем.
- мав низьку вартість реалізації.

Вузький спектр сигналу збільшує швидкість передачі даних по одній лінії (та сама пропускна здатність). Крім того, часто до спектра сигналу пред'являється вимога відсутності постійної складовий, тобто наявності постійного струму між передавачем і приймачем. Зокрема, застосування різних трансформаторних схем гальванічної розв'язки перешкоджає проходженню постійного струму. Використання оптичних каналів дозволяє уникнути надмірного навантаження джерела світла.

Синхронізація передавача і приймача необхідна для того, щоб приймач точно знав, коли зчитувати нову інформацію з лінії зв'язку. Цю проблему складніше вирішити в мережах, ніж під час обміну даними між пристроями, розташованими поблизу один від одного, наприклад між пристроями в комп'ютері або між комп'ютером і принтером. Схема, заснована на використанні окремої тактової лінії зв'язку, добре працює на коротких відстанях, видаляючи інформацію з лінії даних тільки при надходженні тактового імпульсу. У мережах при такому способі виникають проблеми через неоднорідності властивостей провідників в кабелі. Через нерівномірну швидкість поширення сигналу на великі відстані тактові імпульси відповідних сигналів даних можуть надходити занадто пізно або занадто рано, в результаті чого біти даних пропускаються або перераховуються. Ще однією причиною усунення тактових імпульсів в мережі є економія на дорогих кабельних жилах.

Тому в мережах використовуються так звані коди самосинхронізації, у яких сигнал повідомляє передавач, у який момент має відбутися наступний біт (або кілька бітів, якщо код охоплює більше двох станів сигналу). Раптові перепади сигналу (так звані фронти) служать хорошим показником синхронізації приймача і передавача.

При використанні синусоїди як несучого сигналу результуючий код має властивість самосинхронізації, оскільки зміни амплітуди несучої частоти дозволяють приймачу визначити час появи вхідного коду.

Виявлення та виправлення пошкоджених даних важко реалізувати на фізичному рівні, тому це завдання зазвичай обробляється протоколом вищого рівня (канал, мережа, транспорт або програма). Виявлення помилок на фізичному рівні, з іншого боку, економить час, оскільки приймач не чекає, поки кадр буде повністю передано в буфер, а відкидає кадр, як тільки він виявляє біт помилки в кадрі.

Вимоги до запропонованих схем кодування є суперечливими, і кожна з поширених схем цифрового кодування, розглянутих нижче, має переваги та недоліки.

**Потенційний метод кодування**, також відомий як кодування без повернення до нуля (NRZ). Остання назва відноситься до того факту, що при надсиланні послідовності 1 с сигнал не повертається до 0 протягом тактового циклу (інші методи кодування повертатимуться до 0 у цьому випадку, як пояснюється нижче). Метод NRZ простий у реалізації і має високу завадостійкість, але не гарантує самосинхронізацію. Якщо ви надсилаєте довгий рядок з 1 або 0 с, сигнал на дроті не змінюється, тому приймач не може використовувати вхідний сигнал, щоб визначити, коли читати дані знову. Навіть з точними годинниками немає двох осциляторів, які мають однакову частоту. Таким чином, приймач може мати помилки в часі виявлення. Таким чином, при високих швидкостях передачі даних і довгих послідовностях 1 або 0 с невеликі відхилення годинника можуть спричинити помилки в годиннику, що призведе до зчитування неправильних значень бітів.

Іншим істотним недоліком методу NRZ є наявність низькочастотних компонентів, які наближаються до нуля, коли передаються довгі послідовності нулів. Тому чисті коди NRZ в мережі не використовуються. Однак використовуються різні модифікації, які усувають як погану

самосинхронізацію, так і наявність постійних компонентів у коді NRZ. Привабливість кодів NRZ полягає в досить низькій частоті основної гармоніки  $f_0$ ,  $N/2$  Гц. Інші методи кодування, такі як Манчестер, мають вищу основну частоту.

Крім прихованих кодів, мережі також використовують **імпульсне кодування**, коли дані представлені повними імпульсами або їх частинами (ребрами). Найпростішим випадком такого підходу є **біполярний імпульсний** код. У цьому коді одиниці представлені імпульсами однієї полярності, а нулі представлені імпульсами іншої полярності.

Кожен пульс триває половину удару. Такі коди характеризуються хорошими властивостями самосинхронізації, але можуть мати постійні частини, наприклад, при передачі довгих послідовностей або одиниць нуля. Незалежно від того, хто є хто, цей діапазон ширший, ніж діапазон можливих кодів. Таким чином, якщо передаються лише 0 та 1, частота основної частоти коду становитиме  $N$  Гц, що вдвічі більше основної частоти коду NRZ, і частота основної частоти коду АМІ, якщо 1 та 0 відображаються поперемінно. Біполярні імпульсні коди використовуються рідко, оскільки спектр занадто широкий.

Найбільш поширеним методом кодування імпульсів в комп'ютерних мережах є **Манчестерський код**. Використовується в технологіях Ethernet і Token Ring.

Манчестерський код використовує різницю потенціалів або фронти імпульсу для кодування одиниць і нулів. Манчестерське кодування ділить кожну смугу на дві частини. Інформація кодується різницею електричних потенціалів, яка виникає в середині кожного удару. 1 кодується як перехід від низького до високого, а 0 кодується як зворотний перехід. Перепади службового сигналу можуть відбуватися на початку кожного циклу. Він характеризується відмінними властивостями самосинхронізації. Смуга пропускання Манчестерських кодів вужча, ніж інші імпульсні коди. Крім того, немає постійної складової, і частота основної гармоніки в гіршому

випадку (при передачі послідовності 1s або 0s) становить  $N$  Гц, а в найкращому випадку (при передачі чергування 1s і 0s)  $N/2$  дорівнює Гц, такий самий, як імпульсний код NRZ.

### **1.3 Підвищення завадостійкості і інформаційної ємності каналів зв'язку комунікаційних мереж запропонованим методом кодування**

Моніторинг систем управління передбачає передачу інформації по каналу зв'язку комунікаційних мереж систем автоматизації. Якбуло показано в попередніх розділах, в існуючих системах під час передачі інформації використовують коди, що реалізують розрізнювання даних і підвищують швидкість та завадостійкість передачі інформації в каналах зв'язку (наприклад, коди Ріда-Соломона, рангові коди та інш.), а також забезпечують синхронізацію обміну інформацією між абонентами.

Слід зазначити що у таких пристроїв передачі є ряд недоліків а саме: у випадку передачі інформації про ранги відстаней між об'єктами яка необхідна для прийняття рішень в системі управління, обмежується швидкість передачі з причини необхідності передачі по каналу зв'язку додатково до  $m$  кодів об'єктів ще  $m(m-1)/2$  кодів рангів відстаней.

З іншого боку, запропоновані потенціальні (DRP) коди дозволяють значно підвищити пропускну здатність каналу передачі при такому поданні інформації, оскільки кодові слова містять інформацію як про самі стани об'єктів, так і про ранги відстаней між ними. Застосування таких кодів для передачі інформації про стани систем по каналах зв'язку має обмеження, пов'язані із різким збільшенням розрядності DRP-кодів за умов збільшення кількості станів систем, та з особливостями застосування операцій кодування і декодування рангових конфігурацій систем [5].

Для усунення згаданих обмежень і підвищення пропускну здатності каналу передачі в роботі розглядається метод передачі інформації, що використовує додаткові властивості запропонованих кодів. Ці властивості

впливають з того факту, що за своєю природою потенціальні коди є перестановочними кодами з постійною вагою.

Завадостійкість передачі підвищується за рахунок того, що  $DRP$ -код є за своєю природою кодом з постійною вагою (КПВ). Вірогідність  $P_n$  невиявлення однієї помилки в кодовому слові (сумісна вірогідність перетворення одного нуля в одиницю і однієї одиниці в нуль) для КПВ, а отже, і  $DRP$ -коду дорівнює [6]:

$$P_n = \binom{m-1}{1} p(1-p)^{m-2} \binom{n-m+1}{1} p(1-p)^{n-m}, \quad (1.1)$$

де  $m$  - кількість слів у коді. Для рангової конфігурації з чотирьох станів кількість кодованих символів  $m=4$ , кількість рангів відстаней між станами

(розрядність коду)  $n = \frac{m(m-1)}{2} = \frac{4 \cdot 3}{2} = 6$ . Кількість одиниць в  $DRP$ -коді даної конфігурації дорівнює  $(m-1)=3$ . Ймовірність перетворення в кодовому

слові однієї з трьох одиниць в нуль дорівнює  $C_3^1 p_0 (1-p_0)^2$ , а ймовірність перетворення одного з трьох нулів в одиницю складає  $C_3^1 p_0 (1-p_0)^2$ . Користуючись теоремою множення ймовірностей сумісних незалежних подій, отримуємо для  $DRP$ -коду:

$$P_{nDRP} = C_3^1 p_0 (1-p_0)^2 C_3^1 p_0 (1-p_0)^2 = 9 p_0^2 (1-p_0)^4, \quad (1.2)$$

де  $m$  - кількість одиниць у кодовому слові,  $n$  - кількість символів (розрядів) у слові. Визначимо цю характеристику для стандартного КПВ МТА-3 і порівняємо з (1.2). Для КПВ МТА-3, який містить в кодовому слові 3 одиниці і 4 нулі ( $m=3$ ,  $n=7$ ) з формули (1.1) маємо:

$$P_{nKPV} = C_3^1 p(1-p)^2 C_4^1 p(1-p)^3 = 12p^2(1-p)^6. \quad (1.3)$$

Прийнявши значення  $p = 1 \cdot 10^{-4}$ , одержимо для КПВ  $P_{nKPV} \approx 12 \cdot 10^{-8}$ , для *DRP*-коду  $P_{nDRP} \approx 9 \cdot 10^{-8}$ . Оскільки для *DRP*-коду одиниця повинна бути не виявлена одночасно в двох кодових словах, то  $P_{nDRP} \approx 81 \cdot 10^{-16}$ .

Знайдемо відношення

$$\frac{P_{nKPV}}{P_{nDRP}} = \frac{12 \cdot 10^{-8}}{81 \cdot 10^{-16}} \approx 0,15 \cdot 10^8. \quad (1.4)$$

де  $P$  - вірогідність однієї помилки для симетричного каналу.

З виразу (1.8) бачимо, що надійність передачі рангової інформації з використанням *DRP*-коду на декілька порядків вище, чим стандартним КПВ кодом.

Інформаційні характеристики даної системи передачі інформації визначаються кількістю інформації і інформаційна ємністю. Кількість інформації  $I_T$  визначає інформативну здатність джерела і визначається як

$$I_T = \log_2 \left( \frac{m(m-1)}{2}! \right),$$

де  $m$ - кількість кодovаних об'єктів (різних кодових слів). З останнього виразу видно, що кількість переданої інформації зростає з кількістю кодovаних елементів.

Під інформаційною ємністю *DRP* коду  $I_c$  розумітимемо відношення кількості  $Q_r$  прийнятих інформаційних слів до кількості  $Q_r$  переданих кодових слів:

$$I_c = \frac{Q_r}{Q_T} = \frac{m + K_m \cdot n}{m \cdot K_m}, \quad (1.5)$$

де



$$n = \frac{m(m-1)}{2} -$$

кількість рангів в конфігурації, яка і визначає розрядність коду.

Для прикладу, при передачі *DRP*-кодом 30 різних рангових конфігурацій чотиривимірного симплексу треба передати слів, з яких можна вибрати  $Q_r = 4 + 30 \cdot 6 = 184$  інформаційних слова, звідки  $I_c = 184/120 \approx 1,53$ .

Коефіцієнт підвищення пропускної здатності каналу різко збільшується із збільшенням кількості станів системи, про які передається інформація (це може бути, наприклад, інформація про просторову конфігурацію цілей противника, про розміщення мобільних абонентів в бездротовій мережі, про стан нейронної мережі та багато інш.). Наприклад, для системи з 4 станами пропускна здатність каналу підвищується порівняно з відомими методами передачі в 10 раз, для системи з 10 станами – в 55 раз.

#### **1.4 Підвищення ефективності прийняття управлінських рішень за рахунок запропонованого методу опису і кодування даних**

Основна ідея використання запропонованого методу для представлення та кодування інформації про стани системи полягає в описі цих станів у різних параметричних просторах та інтеграції алгоритмів прийняття рішень за умови підвищення їх ефективності. Отже, проаналізуємо переваги. Пропонований спосіб є кращим від відомих на даний момент способів.

Розглянемо розв'язання даної проблеми для такої системи прийняття рішень, як розпізнавання образів [13-16]. Результати при цьому можна розповсюдити на всі системи прийняття рішень, оскільки викладки при цьому будуть відрізнятися тільки видом елементів кодування – для систем управління ними будуть стани систем, для нейронних мереж – нейрони і т.і.

Проблема вибору способу кодування в системах розпізнавання мови виникає в тих випадках, коли наслідком роботи нижніх рівнів ієрархії є

сегментація неперервного мовного сигналу в послідовність дискретних звукотипів  $S_j$ , об'єднаних в стрічки  $W_1 = ( S_1, S_2, \dots, S_j, \dots, S_q )$ . Звукотипи в запам'ятовуючому пристрої представляються у вигляді двійкових кодів, при цьому кодування виконується з урахуванням вимоги їх розрізнення. Щоб зберігати інформацію про просторову конфігурацію звукотипів, необхідно запам'ятовувати крім  $m$  кодів звукотипів ще і  $m*(m-1)/2$  кодів відстаней між звукотипами. Процес знаходження відстані між рядками символів, наряду з виконанням арифметичної операції додавання відстаней між парами символів з еталонної і стрічки, що розпізнається, потребує кожен раз звернення до пам'яті для зчитування чисел, які характеризують ці відстані.

Очевидно, що загальноприйнятий підхід до кодування має ті недоліки, що потребує додаткових апаратних затрат на зберігання  $m*(m-1)/2$  кодів відстаней, а також значно обмежує швидкість розпізнавання через втрати часу на зчитування з пам'яті цих відстаней. В зв'язку з цим нами запропонований такий спосіб двійкового представлення компонент векторів  $w_i, w_j$ , при якому інформація про відстані між компонентами  $s_k^i$  і  $s_k^j$  векторів  $w_i$  і  $w_j$  містилась би в їх кодах. При цьому величина відстані визначається безпосередньо в арифметико-логічному пристрої шляхом порівняння кодів за допомогою операції “сума по модулю 2”.

Оцінимо переваги обраного способу кодування в порівнянні з відомими.

Відомий спосіб розпізнавання образів і пристрою, що його реалізує, відповідно до якого сприймають пред'явлений образ за допомогою датчика, роблять його перетворення, виділяють характерні ознаки і виконують опис, після чого роблять розпізнавання пред'явленого образу по максимуму кореляційної функції з одним з еталонних [17].

Недоліком даного способу є обмеження, що накладається на обсяг розпізнюваного словника, тому що для збереження еталонного опису образу у вигляді масиву ознак потрібно великий обсяг пам'яті.

Відомий метод розпізнавання образів передбачає сприйняття неперервного образу, його трансформацію, виділення характерних ознак, дискретизацію послідовності елементів, сегментне маркування елементів, створення двійкового опису розпізнаного рядка символів у вигляді відмінних для різних символів двійкових кодів. Розпізнавання виконується на основі мінімальної відстані до одного з еталонних образів. Для цього зчитуються з пам'яті коди відстаней між кожною порівнюваною парою символів з розпізнаного та еталонного рядків і обчислюються їхні суми.

Однак, серед недоліків цього методу можна відзначити потребу в додатковому обсязі пам'яті для збереження кодів відстаней, а також обмежену швидкість розпізнавання через втрати часу на зчитування цих кодів з пам'яті.

Технічно найбільш подібним до запропонованого рішення є метод розпізнавання образів, в якому спочатку сприймається неперервний образ, його перетворюють, перетворюють у послідовність елементів, створюють двійковий опис розпізнаваної послідовності у вигляді двійкових кодів, які відзначають відмінності. Потім відбувається класифікація, базуючись на мінімальній відстані до одного з еталонних рядків. Це досягається шляхом логічного порівняння розпізнаваної та еталонної послідовностей елементів, визначення відстані Хеммінга між ними, і прийняття рішення щодо образу на основі отриманих результатів [19].

Недоліками цього методу є обмеження об'єму словника розпізнаваних образів, яке виникає внаслідок можливості виникнення помилок через властивості кодів, що утримують відмінності. Також є обмеження в продуктивності через потребу в обчисленні відстані Хеммінга. Наприклад, якщо символи образу позначаються як "цілі" елементи, то для двійкового коду, що утримує відмінності, виконуються такі умови:

$$\exists |i-j| \leq t \text{ впливає } H(D_i, D_j) = |i-j|, \quad (1.6)$$

$$\exists |i-j| < t \text{ впливає } H(D_i, D_j) \neq |i-j|, \quad (1.7)$$

де  $i, j \in \{1, 2, \dots, N\}$  - цілі числа (символи елементів);  $t$  - поріг, заданий цілим числом;  $D_i, D_j$  - двійкові коди елементів  $i, j$ , що скорочують різницю;  $H(D_i, D_j)$  - відстань Хеммінга між двома кодовими словами. При цьому для всякої трійки образів  $\{O_{k1}, O_{k2}, O_{k3}\}$ , описаної у виді рядків символів

$$\begin{aligned} O_{k1} &= j_1 + 1, j_2 + 1, \dots, j_m + 1; \\ O_{k2} &= j_1, j_2, \dots, j_m; \\ O_{k3} &= j_1 - 1, j_2 - 1, \dots, j_m - 1, \end{aligned} \quad (1.8)$$

Хеммінгові відстані  $H(O_{k1}, O_{k2})$  і  $H(O_{k2}, O_{k3})$  рівні між собою. Зазначена невизначеність приводить до помилок у розпізнаванні при наявності в словнику образів із зазначеними описами.

Метою застосування результатів магістерської роботи в пристроях розпізнавання образів автоматизованих систем управління є збільшення обсягу словника розпізнаваних образів і підвищення швидкодії розпізнавання. Для цього за запропонованим способом двійковий опис розпізнаваної послідовності елементів формується у вигляді двійкових кодів, що зберігають ранги відстаней між елементами, а для виконання класифікації роблять логічне порівняння розпізнаваної й еталонної послідовностей елементів.

Двійковим кодом  $\beta$ , що зберігає ранги відстаней (DRP - кодом) є відображення  $i \rightarrow \beta_i$  множини  $m = \{1, 2, \dots, m\}$  у множину  $\{0, 1\}^n$  двійкових послідовностей довжини  $n$  таке, що

$$\forall_{i, j \in m} (R(d_{ij}) = k \Rightarrow R(h_{ij}) = k), \quad (1.9)$$

де  $R(d_{ij})$  - ранг відстані  $d_{ij}$  між елементами  $i$  та  $j$  у просторі елементів,  $R(h_{ij})$  - ранг відстані  $h_{ij}$  у просторі двійкових кодів,  $k$  - ціле число, величина рангу. Можна довести, що кодування послідовності елементів, що описує розпізнаваний образ, двійковим  $RP$  - кодом є умовою адекватності розпізнавання образів на просторі двійкових рядків (доказ опускається).

При виконанні класифікації за відомим способом при логічному порівнянні розпізнаваної й еталонної послідовностей елементів необхідно визначати відстань Хеммінга між ними, що знижує швидкість зазначеного способу розпізнавання образів. У запропонованому способі визначати відстань Хеммінга не потрібно, тому процес розпізнавання відбувається швидше. Крім того, формування двійкового опису розпізнаваної послідовності елементів у виді двійковий кодів, що зберігають ранги відстаней між елементами, запобігає від виникнення помилок класифікації в залежності від порядку проходження елементів у послідовності, що описує образ, що дозволяє розширити словник розпізнаваних образів.

Сутність запропонованого способу пояснюється прикладом.

Нехай алфавіт  $A$  елементів, з яких складаються розпізнавані образи, містить п'ять елементів, що умовно позначимо цифрами від 1 до 5 :  $A = \{1, 2, 3, 4, 5\}$ .

Двійковим кодом  $D$ , що зберігає різниці, буде множина двійковий кодів

$$D = \{0000, 0001, 0011, 0111, 0110\}. \quad (1.10)$$

Двійковим кодом  $V$ , що зберігає ранги відстаней між елементами, може бути, наприклад, множина двійковий кодів :

$$V = \{0011, 0010, 0001, 0111, 1011\}. \quad (1.11)$$

Вид двійкового  $RP$ -коду залежить від просторової конфігурації елементів. Приклад конфігурації, що може бути представлена двійковим  $RP$ -

кодом, наведенов формулі (1.12), Якщо образи ПРО<sub>1</sub>, ПРО<sub>3</sub> представлені послідовностями елементів ПРО<sub>1</sub>=(1,2,3), ПРО<sub>3</sub>=(3,4,5), а пред'явлена реалізація образу описується послідовністю елементів ПРО<sub>2</sub>=(2,3,4), то їхній двійкове представлення D- кодом буде мати вигляд:

$$\begin{aligned}
 \mathbf{O}_1 &= \overbrace{0000}^2 \quad \overbrace{0001}^3 \quad \overbrace{0011}^4 \\
 \mathbf{O}_2 &= \overbrace{0001}^3 \quad \overbrace{0011}^4 \quad \overbrace{0111}^5 \\
 \mathbf{O}_3 &= \overbrace{0001}^3 \quad \overbrace{0111}^4 \quad \overbrace{1011}^5
 \end{aligned} \quad (1.12)$$

Відстань Хеммінга для даних послідовностей:

$$H(\mathbf{O}_1^\nabla, \mathbf{O}_2^\nabla) = H(\mathbf{O}_2^\nabla, \mathbf{O}_3^\nabla) = 3 \quad (1.13)$$

Тобто образи ПРО<sub>1</sub> і ПРО<sub>3</sub> стають нерозрізнюваними стосовно реалізації ПРО<sub>2</sub>, що може привести до помилки класифікації.

Двійкове представлення образів RP- кодом буде мати вид:

$$\begin{aligned}
 \mathbf{O}_1^* &= \overbrace{0011}^2 \quad \overbrace{0010}^3 \quad \overbrace{0001}^4 \\
 \mathbf{O}_2^* &= \overbrace{0010}^3 \quad \overbrace{0001}^4 \quad \overbrace{0111}^5 \\
 \mathbf{O}_3^* &= \overbrace{0001}^3 \quad \overbrace{0111}^4 \quad \overbrace{1011}^5
 \end{aligned}$$

Відстані S, отримані при логічному порівнянні шляхом підсумовування за модулем двіох двійкових послідовності, будуть рівні :



$$\begin{aligned} S(\mathbf{O}_1^*, \mathbf{O}_2^*) &= 0001 \quad 0011 \quad 0110 \\ S(\mathbf{O}_2^*, \mathbf{O}_3^*) &= 0011 \quad 0110 \quad 1100, \end{aligned} \quad (1.14)$$

при цьому  $S(\mathbf{O}_2^*, \mathbf{O}_3^*) > S(\mathbf{O}_1^*, \mathbf{O}_2^*)$ , тому пред'явлена реалізація образу ПРО<sub>2</sub> може бути розпізнана лише як образ ПРО<sub>1</sub>.

З наведеного прикладу видно, що формування двійкового опису розпізнаваної послідовності елементів у виді двійковий кодів, що зберігають ранги відстаней між елементами, усуває обмеження на порядок проходження елементів послідовності, що описує образ. Це дозволяє розширити словник розпізнаваних образів до розмірів, обмежених лише технічними можливостями розпізнавального пристрою і незалежних від способу розпізнавання образів.

Для класифікації образу за запропонованим методом не потрібно обчислювати відстань Хеммінга після логічного порівняння розпізнаної та еталонної послідовностей елементів, як це необхідно у відомому методі. Підрахунок кількості одиниць у двійковій послідовності, яка визначає відстань Хеммінга, може бути виконаний апаратно або програмно. У випадку апаратної реалізації тимчасові витрати на визначення Хеммінгової відстані визначаються таким виразом:

$$T_x = t \cdot n \cdot m \cdot p, \quad (1.15)$$

де  $t$  - тривалість тактового періоду ;

$n$  - розрядність двійкового D- коду;

$m$  - число елементів у послідовності;

$p$  - кількість образів у словнику розпізнаваних образів.

Виграш  $\delta$  у швидкодії розпізнавання по запропонованому способі може бути охарактеризований скороченням часових витрат на обчислення Хеммінгових відстаней:

$$\delta = \frac{T_u}{T_n} \cdot 100\% = \frac{T_u}{T_u - T_x} \cdot 100\% = \frac{T_u}{T_u - t \cdot n \cdot m \cdot p} \cdot 100\%, \quad (1.16)$$

де  $T_n$  - час розпізнавання по запропонованому способі;

$T_u$  - час розпізнавання за відомим способом.

Наприклад, для випадку розпізнавання мовних образів акустичний сигнал, що відповідає одному слову, може бути перетворений у послідовність, що складається в середньому з 10 фонемоподібних елементів, а число розпізнаваних образів у словнику в середньому складає 200 слів, при цьому середня тривалість слова складає приблизно 0,8 сек. Розрядність DRP-коду для кодування алфавіту А, що містить 40-60 фонемоподібних елементів, визначається співвідношенням  $n \geq 7$ . Якщо час розпізнавання по відомому способі  $T_u$  прийняти рівним приблизно реальному часу пред'явлення мовного образу,  $T_u \approx (1 \div 1,2) T_p$ , що характерно для найбільш досконалих пристроїв розпізнавання мови, а тривалість  $t$  тактового періоду прийняти рівну 2 мксек. (що справедливо, наприклад, для мікропроцесорів середньої продуктивності), то вигравш у швидкодії чисельно дорівнює:

$$\delta = \left( 1 - \frac{2 \cdot 10^{-6} \cdot 7 \cdot 10 \cdot 200}{0,8} \right) \cdot 100\% \approx 104\%.$$

#### 1.4 Висновки до розділу

Аналіз різних варіантів відомих методів опису та кодування інформації для передачі в комунікаційних мережах систем автоматизованого управління показав, що враховуючи такі вимоги, як розрізнюваність кодованих елементів даних, можливість виявлення та корекції помилок, що виникають у каналі зв'язку під час передачі інформації, а також синхронізація обміну даними між абонентами мережі, ці методи не мають властивостей зберігання інформації про відношення відстаней між станами об'єктів управління. Таким

чином, вони не забезпечують можливості ідентифікації станів та прийняття рішень щодо управління на основі цієї інформації.

Порівняльний аналіз запропонованого методу опису інформації і її кодування показав, що його застосування в різноманітних системах прийняття рішень і управління дозволяє підвищити ефективність їх роботи, оскільки дозволяє уніфікувати алгоритми прийняття рішень, підвищити їх швидкість їх прийняття незалежно від використаного способу опису об'єктів управління в параметричних просторах. Тому розробка теоретичного підґрунтя і технічних засобів для реалізації запропонованих методів, що будуть вирішуватися в ході досліджень в даній роботі, є доцільними з точки зору економічної ефективності.

## 2 РОЗРОБКА І МАТЕМАТИЧНЕ ОБГРУНТУВАННЯ ЗАПРОПОНОВАНИХ МЕТОДІВ ТА ЗАСОБІВ ОПИСУ І ПЕРЕДАЧІ ІНФОРМАЦІЇ

### 2.1 Математичне обґрунтування рангового опису інформації

Як показано в попередньому розділі дослідження, загальноприйнятий метод кодування має недолік у тому, що вимагає додаткових витрат на обладнання для зберігання кодів відстані  $m_h^*(m_h-1)$ . Крім того, втрата часу значно обмежує швидкість передачі інформації. Ці відстані призначені для передачі через канал. У зв'язку з цим нами запропоновано наступний спосіб двійкового представлення векторних компонент (станів системи)  $w_i, w_j$ . У цьому методі код містить інформацію про відстань між компонентами  $s_k^i$  і  $s_k^j$  векторів  $w_i$  і  $w_j$ . При цьому значення відстані визначається безпосередньо в арифметико-логічному блоці шляхом порівняння кодів за посиланням «I».

Означення 1. Ранговий склад  $\mathfrak{R}$  простору  $m$  об'єктів – це набір базових підмножин  $(m-1)$  - елементи цих підмножин є рангами відстаней до тих самих об'єктів.

Конфігурацію рангу можна отримати, впорядкувавши відстані між об'єктами в просторі параметрів у порядку зростання та призначивши номери рангу.

Визначення 2. Код  $v$  збереження діапазону відстані (код *DRP*-код, інакше потенційний код) — це відображення  $i \rightarrow B_i$  множини  $M = \{1, 2, \dots, m\}$  в множину  $\{0, 1\}^n$  двійкової послідовності довжиною  $n$ .

$$\forall_{i,j} (R(d_{ij}) = r \Rightarrow R(h_{ij}) = r), \quad r = \overline{1, m_r}, \quad i, j \in M \quad (2.1)$$

У рівнянні (2.1)  $R(d_{ij})$  ранг відстані  $d_{ij}$  між об'єктом  $i$  і  $j$  у просторі

об'єктів;  $R(h_{ij})$  – ранг відстані  $h_{ij}$  в просторі двійкового коду;  $r$  - ціле, конкретне значення рангу;  $m_r$  - максимальний ранг. Надалі під кодованими об'єктами будуть розумітися елементи системи.

Наведене визначення 2 використовує код *DRP* для метричних просторів (станів системи) з певними строгими зв'язками впорядкування на наборі лінійно впорядкованих відстаней між об'єктами або просторами з нестрогими зв'язками впорядкування.

Його можна інтерпретувати як двійкове представлення. Колони на однаковій відстані вдаряються в один і той же предмет. Потенційний код може бути розроблений для двох різних програм. Перша з них відноситься до завдань, які ідентифікують стан системи (наприклад, завдання кластеризації), а друга стосується таких завдань, як розпізнавання рядків, передача інформації про захист від перешкод тощо. У першому випадку виконується аксіома тотожності  $R(d_{ii})=0$ , оскільки відстані між рівними точками не обчислюються, тому ранжування відстаней між кодами країн не потрібне. У цьому обговоренні ранг відстані  $R(h_{ij})=0$  між двійковими словами  $b_i$  і  $b_j$  *DRP* - коду отримується за допомогою операції мультиплікативної логіки множення AND та визначається за такою формулою:

$$R(h_{ij}) = \log_2(b_i \wedge b_j), \quad (2.2)$$

Тут операція AND позначається символом  $\wedge$ .

У другому випадку вам потрібно створити свій код з урахуванням операції додавання XOR («виключне АБО»).

Розглянемо модель конфігурації рангу, необхідної для визначення властивостей прихованого коду в першому випадку. Для їх представлення були розроблені геометричні, алгебраїчні, комбінаторні та топологічні моделі. Багатовимірні симплекси можна класифікувати як геометричні симплекси, а графи можна класифікувати як топологічні симплекси. На рисунку 2.1 показано структуру рангів у тривимірному вигляді та у вигляді графіка.

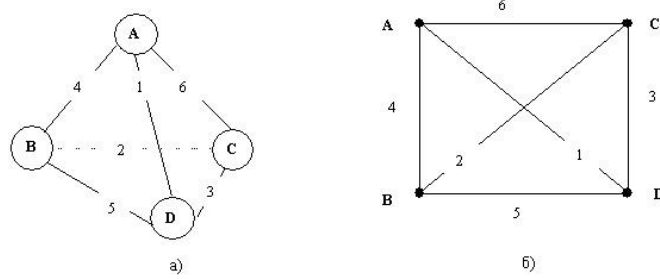


Рисунок 2.1 - Геометричні моделі рангової конфігурації: а – тривимірний симплекс; б–повний регулярний граф

Візуалізуйте композицію рангів за допомогою симплексів і графіків.

На малюнку 2.1 вершини А, В, С і D вказують на закодовані об'єкти, а числа на кінцях симплекса та дуги на діаграмі вказують на їхні інцидентні ранги.

Алгебраїчна модель дозволяє компактно описати рангову конфігурацію згідно з визначенням 1 у вигляді набору рангових підмножин. наприклад, рангову конфігурацію рис.2.1 можна записати в алгебричному вигляді як

$$K_4 = \{ \{1, 4, 6\}, \{2, 4, 5\}, \{2, 3, 6\}, \{1, 3, 5\} \}. \quad (2.3)$$

Ранги

	6	5	4	3	2	1
А	1	0	1	0	0	1
В	0	1	1	0	1	0
С	1	0	0	1	1	0
Д	0	1	0	1	0	1

а)

	А	В	С	Д
А		4	6	1
В	4		2	5
С	6	2		3
Д	1	5	3	

б)

Рисунок 2.2 - Комбінаторні моделі рангової конфігурації: а) – матриця інцидентностей графа рис.2.1, б; б) – матриця суміжностей рангів для графа рис.2.1,б



Комбінаторні моделі конфігурацій можна представити у вигляді діаграм і блок-схем, які широко використовуються в теорії кодування [5], а також у вигляді перестановок матриць суміжності рангів графів. Детальне вивчення цієї схеми та її блок-схемних варіантів показало, що вона не підходить для моделювання рангових конфігурацій, оскільки не виконується одна з умов, що характеризує зручність використання такої моделі [3,4]. Тому ми використали матрицю входжень графіка для моделювання комбінаторних аспектів. (рисунок 2.2, а) і рангову матрицю суміжності цього малюнка (рисунок 2.2, б).

Ці комбіновані моделі дозволяють нам визначити фундаментальні властивості основного коду, такі як бітрейт і цілісність.

З матриці входжень 2а) ми можемо відразу побачити, що потенційна швидкість коду  $n$  дорівнює кількості рангів, припускаючи, що ранги визначаються логічною операцією «І» дорівнюватиме кількості рангів:

$$n = \frac{m(m-1)}{2}, \quad (2.4)$$

де  $m$  - кількість кодованих об'єктів. Наприклад, для  $m=4$   
 $n = \frac{4(4-1)}{2} = \frac{12}{2} = 6$ , для  $m=5$  маємо  $n=10$  і т.д.

Використання матриці інцидентності також дає можливість створити алгоритм кодування об'єктів DRP, що включає два етапи: :

- 1) Заповнення матриці розміром  $m \times n$  нулями;
- 2) Перегляд матриці по стовпцях (рангах) і установка одиниці в тих стрічках, що відповідають об'єктам, інцидентним цим рангам.

На рис. 2.2,а рядки матриці суміжності графа утворюють кодові слова шуканого для заданої рангової конфігурації  $DRP$ -коду. Як приклад визначимо ранг відстані між кодовими словами символів А і D, для чого скористаємося виразом (2.2):

$$R(h_{AD}) = \log_2(b_A \wedge b_D) = \log_2(101001 \wedge 010101) = \log_2 2^1 = 1.$$

На практиці операція логарифмування для визначення точного рангу не потрібна, оскільки відстані між кодовими словами ранжуються операцією AND.

З моделі рис.2.2, а також видно, що за своєю формою *DRP*-код є перестановочним з постійною вагою. Виходячи з цього, для визначення його перешкодостійкості можемо скористатися відомими [17] для цих кодів залежностями. Вірогідність  $P_n$  невиявлення однієї помилки в кодовому слові (сумісна вірогідність перетворення однієї одиниці в нуль і одного нуля в одиницю) для *DRP*-коду дорівнює:

$$P_n = \binom{m-1}{1} p(1-p)^{m-2} \binom{n-m+1}{1} p(1-p)^{n-m}, \quad (2.5)$$

де  $m$  - кількість слів у коді. Користуючись теоремою множення ймовірностей сумісних незалежних подій, отримуємо для *DRP*-коду:

$$P_{nDRP} = C_3^1 p_0(1-p_0)^2 C_3^1 p_0(1-p_0)^2 = 9p_0^2(1-p_0)^4, \quad (2.6)$$

де  $m$  - кількість одиниць у кодовому слові,  $n$  - кількість символів (розрядів) у слові. Визначимо цю характеристику для стандартного КПВ МТА-3 і порівняємо з (2.6). Для КПВ МТА-3, який містить в кодовому слові 4 нулі і 3 одиниці ( $m=3$ ,  $n=7$ ) з формули (2.5) маємо:

$$P_{nKPV} = C_3^1 p(1-p)^2 C_4^1 p(1-p)^3 = 12p^2(1-p)^6. \quad (2.7)$$

Прийнявши значення  $p = 1 \cdot 10^{-4}$ , одержимо для КПВ  $P_{nKPV} \approx 12 \cdot 10^{-8}$ , для *DRP*-коду  $P_{nDRP} \approx 9 \cdot 10^{-8}$ . Оскільки для *DRP*-коду одиниця повинна бути не виявлена одночасно в двох кодових словах, то  $P_{nDRP} \approx 81 \cdot 10^{-16}$ .

Знайдемо відношення

$$\frac{P_{n\text{КПВ}}}{P_{n\text{DRP}}} = \frac{12 \cdot 10^{-8}}{81 \cdot 10^{-16}} \approx 0,15 \cdot 10^8.$$

Тобто, надійність передачі рангової інформації з використанням DRP-коду на декілька порядків вище, чим стандартним КПВ кодом.

Модель рангової конфігурації у вигляді матриці суміжності рангів дозволяє дійти висновку, що множину конфігурацій можна представити групою на наборі перестановок. Ця характеристика дозволяє не лише кодувати символи, але й конфігурації, зберігаючи інформацію про них у компактованій формі.

Потужність множини конфігурацій  $(m-1)$  - вимірного симплекса дорівнює:

$$K_m = \frac{(m(m-1)/2)!}{m!} = \frac{n!}{m!}. \quad (2.8)$$

Для об'єктів, які мають мітки, кількість рангових конфігурацій визначається потужністю множини перестановок рангів, і це не вимагає окремого доведення:

$$K_m = n!. \quad (2.9)$$

Збільшення кількості конфігурацій при зростанні числа символів  $m$  відповідає явищу, що можна охарактеризувати як "комбінаторний вибух". Наприклад, при  $m = 4$  кількість рангових конфігурацій  $K_4$  непоміченого симплексу відповідно до виразу (2.8)

$$K_4 = \frac{6!}{4!} = \frac{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6}{1 \cdot 2 \cdot 3 \cdot 4} = 30,$$

для п'яти символів  $K_5 = 30240$ , для шести символів  $K_6 \approx 1,8 \cdot 10^9$ . Ще швидше зростає кількість конфігурацій для поміченого симплексу, що видно з виразу

(2.9), та свідчить про велику інформаційну потужність *DRP*-коду в той час коли користувач має потребу в інформації про відносини між елементами даних.

Якщо передавати не лише рангову конфігурацію, а її код, який відповідає номеру рангової перестановки, то пропускну спроможність каналу можна збільшити. Завдяки цьому можна забезпечити секретність коду  $C_R$  вибором одного з можливих порядків перестановки рангів на матриці суміжності і визначиться вірогідністю вгадування номера цього порядку:

$$C_R = \frac{1}{n!}. \quad (2.10)$$

Адекватність моделей рангових конфігурацій, застосованих до визначення характеристик потенціальних точок, доведена аналітично, тому проведення експерименту з даної метою буде надлишковим.

Для доведення повноти кодів в другому випадку (для тих застосувань, які вимагають існування аксіоми ідентичності), розроблена топологічна інтервальна модель рангової конфігурації, приклад якої наведений на рис.2.3.

Дана модель представляє собою лінійний відрізок довжиною  $2^q$ , де  $q$ -вибрана розрядність коду, поділений на одиничні інтервали, границі інтервалів відповідають цілим числам з діапазону  $[0 \dots 2^q]$ . На цьому відрізку укладені інтервали, які в промасштабованому вигляді відповідають рангам конфігурації. Для моделі справедливими є такі аксіоми.

Аксіоми різницевої операції:

- аксіома симетрії;

$$\forall_{i,j \in M} (b_i \oplus b_j) = (b_j \oplus b_i) = |b_i \oplus b_j| = h_{ij}; \quad (2.11)$$

- аксіома оберненості операції.

$$b_i \oplus h_{ij} = b_j; \quad b_j \oplus h_{ij} = b_i . \quad (2.12)$$

Аксіоми рангових інтервалів:

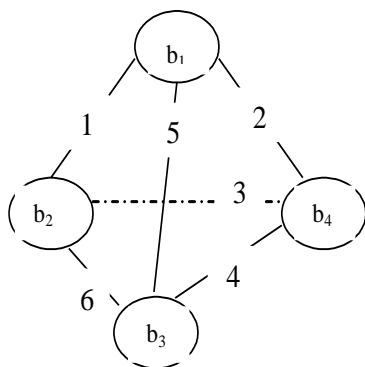
- аксіома жорсткості інтервалів в циклі

$$h_{ij}(r_k) + h_{ij}(r_{k+1}) = h_{ij}(r_k + r_{k+1}); \quad (2.13)$$

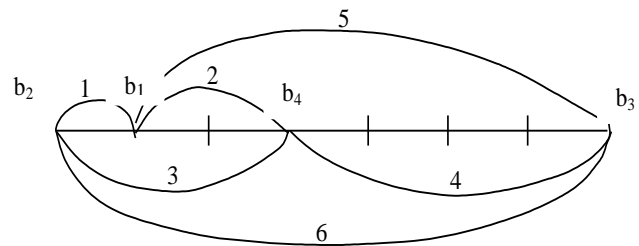
- аксіома трикутника для рангів в циклі

$$r_k + r_{k+1} \geq r_c ; \quad (2.14)$$

де  $r_k$  і  $r_{k+1}$  - ранги  $k$ -го і  $k+1$ -го інтервалів на моделі (ранги сусідніх інтервалів),  $r_c$  - ранг інтервалу, що замикає. Дані ранги позначені дугами на топологічному графові моделі (рис. 2.3,б). Поняття циклу на інтервальній моделі відповідає поняттю циклу на топологічному графові.



a)



$$K_4 = \{\{1,2,5\}, \{1,3,6\}, \{2,3,4\},$$

б)

Рисунок 2.3 - Інтервальна модель рангової конфігурації

На основі інтервальної моделі можна зробити висновок, що допустимими є тільки ті рангові конфігурації, для яких виконуються аксіоми (2.13) і (2.14).

Як впливає з теореми, інтервальна модель тільки показує, що кількість рангових конфігурацій за умови використання операції XOR є обмеженою, що не дозволяє побудувати повний потенціальний код. Але вона не дає аналітичного виразу для визначення кількості можливих конфігурацій і методу побудови потенціального коду. Для оцінки адекватності цієї моделі був розроблений алгоритм, що базується на повному переборі всіх можливих сполучень з  $m$  по  $2^q$  кодів цілих двійкових чисел розрядністю  $q$  на інтервалі  $[0 \dots 2^q]$ . Проведений в роботі машинний експеримент підтвердив, що DRP-код у цьому випадку є неповним. Наприклад, кількість дозволених рангових конфігурацій для 3-х вимірного симплекса (4 кодованих об'єкти) дорівнює 7 із 30 можливих, і коефіцієнт повноти коду відповідно дорівнює  $7/30$ , тобто приблизно 23,3%.

Тому автори пропонують у цьому випадку використовувати потенціальний код, спроектований на базі використання модифікованої логічної операції AND “з блокуванням за порогом 1”. Вона складається з логічної операції AND над потенціальними кодами об'єктів, отриманих алгоритмом 1, з наступним виконанням операції XOR над результатом. Якщо сума одиниць в результаті більша 1, то результат блокується до нуля, що дає можливість забезпечити виконання аксіоми ідентичності при визначенні рангу відстані між одними і тими ж об'єктами.

Для реалізації даної операції автором розроблено схему логічного елемента “І з самоблокуванням”, яка використовується в якості декодера DRP-коду 9 в схемі пристрою передачі інформації на рисунку 2.5. Функціональна схема даного логічного елемента представлена на рисунку 2.4.

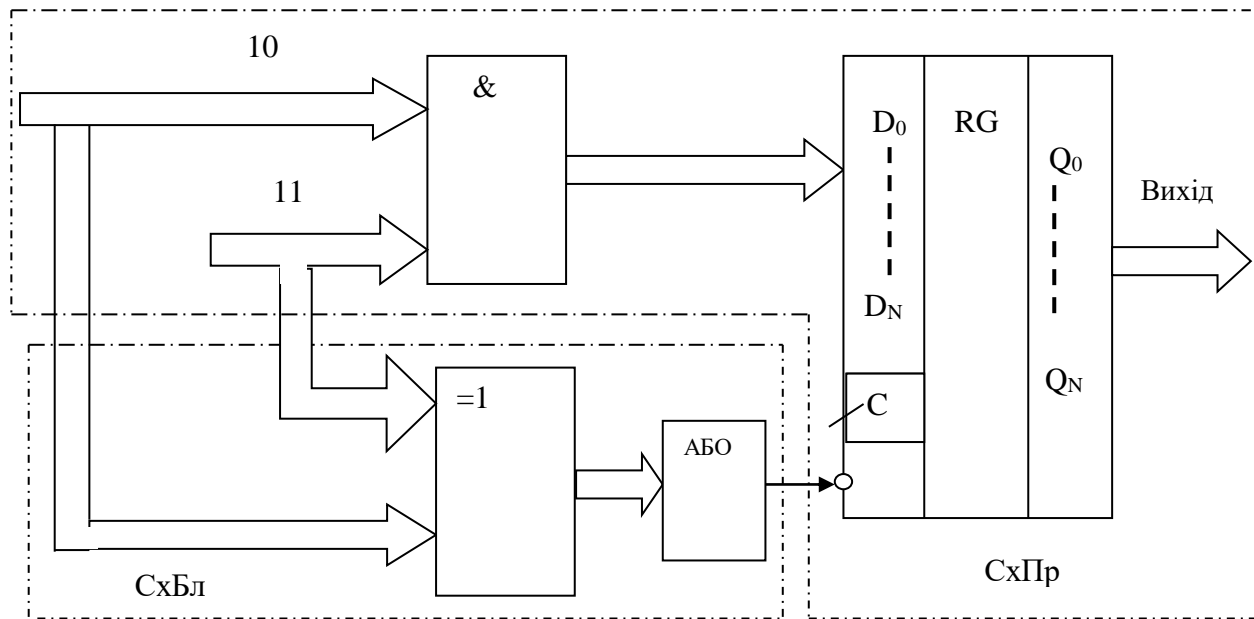


Рисунок 2.4 – Структурна схема декодера рангового коду

## 2.2 Розробка структурної схеми пристрою передавання інформації

Основна мета проекту полягає у створенні пристрою для передачі інформації, який здатний представляти дані про рангову конфігурацію об'єктів у будь-якому параметричному просторі. Це досягається за допомогою введення нових елементів та зв'язків шляхом кодування DRP-кодом. Такий підхід дозволяє уникнути обмежень на кількість об'єктів у ранговій конфігурації, яка підлягає кодуванню, і підвищити завадостійкість передачі інформації.

Можливість усунення обмежень на кількість об'єктів у ранговій конфігурації та збільшення завадостійкості передачі інформації досягається через використання джерела пристрою для обчислення рангів відстаней між об'єктами у просторі координат як кодера. Також використовується штучна нейронна мережа (ШНМ) у випадку, коли стан об'єктів описується характерними параметрами в поєднанні з кодером каналу, який здійснює кодування рангової конфігурації DRP-кодом.

За рахунок того, що за своєю природою DRP-код є кодом з постійною вагою (КПВ) підвищується завадостійкість передачі. Як було попередньо показано, запропонований спосіб передачі інформації збільшує завадостійкість порівняно з аналогом на порядок і вище.

Поставлена задача вирішується тим, що інформація з джерела повідомлень кодується кодером джерела інформації в рангову конфігурацію, причому на вхід цього кодера під'єднується штучна нейронна мережа для отримання функції відстаней між кодованими елементами, після чого за допомогою кодера каналу кодується в DRP-код. На наступному етапі перетворюється в інформаційний сигнал в модуляторі, далі сигнал передається по каналу передачі, на виході каналу передачі сприймається приймачем сигналу, демодулятором демодулюється в кодовий сигнал, і перетворюється в рангову конфігурацію за допомогою декодера DRP-коду, яка використовується безпосередньо системою прийняття рішень для управління.

На рисунку 2.5 представлена структурна схема пристрою для реалізації способу передачі інформації.

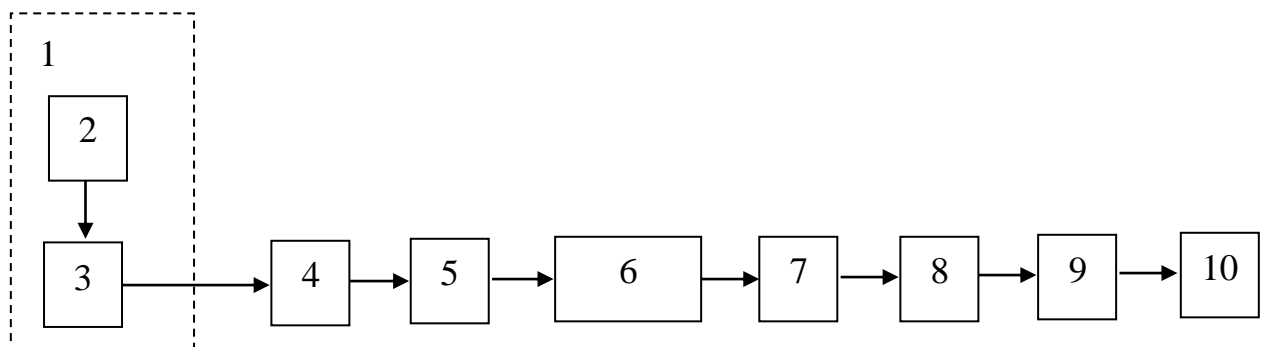


Рисунок 2.5 – Структурна схема пристрою передачі інформації

Пристрій вміщає джерело інформації 1, в нього входить джерело повідомлень 2 (група елементів або об'єктів) і кодер джерела інформації 3, джерело з'єднано з кодером каналу 4, даний канал під'єднано до входу модулятора сигналу 5, вихід якого в свою чергу з'єднано з входом каналу



передачі інформації 6. Вихід каналу передачі інформації під'єднаний до входу приймача сигналу 7, вихід якого в свою чергу з'єднано з входом демодулятора сигналу 8, його підключено до декодера DRP-коду 9 та послідовно з'єднано з блоком прийняття рішень 10.

У ролі кодера джерела може застосовуватися штучна нейронна мережа (ШНМ) для випадку, коли стан об'єктів описується не координатами, а характерними параметрами (ознаками), або пристрій обчислення рангів відстаней між об'єктами у просторі координат.

Принцип роботи пристрою виглядає так: джерело інформації 1 з подальшими двома блоками складає єдине ціле, тому в ньому інформація з джерела повідомлень 2, надходить на кодер джерела інформації 3, де її перетворюють в рангову конфігурацію. У разі потреби під'єднується штучна нейронна мережа для отримання функції відстаней між кодованими елементами. Після чого з допомогою кодера каналу 4, отриману рангову конфігурацію кодують DRP-кодом. Далі на модулятор 5 надходить кодова послідовність, модулятор здійснює його пребудову в інформаційний сигнал, який переміщується по каналу передачі 6. Отже, в результаті виконаних операцій інформація переміщується до приймача сигналу 7 на виході каналу передачі, з нього вона потрапляє на демодулятор 8, який в кодову послідовність здійснює перетворення інформаційного сигналу. Отриманий код декодується декодером DRP-коду 9.

За допомогою порозрядної логічної операції “Г”, під час декодування визначаються, ранги відстаней між станами системи, закодованими DRP-кодом, однак при цьому є вірогідність виникнення помилок. Це зумовлено тим, що під час визначення відстані між однаковими марками не виконується аксіома ідентичності. Згідно цього було запропоновано модифікувати дану логічну операцію таким чином, щоб для неї виконувалась аксіома ідентичності. Для цього була розроблена відповідна логічна схема, яка скидає результат порівняння однакових марок до нуля та сформульовано

математичний опис операції. Модифікована операція була названа логічною операцією ““Г” з самоблокуванням”.

В результаті на блок прийняття рішень 10 надходить отримана інформація.

## **2.3 Опис моделі кодера рангової інформації**

У випадку, коли стани системи описані у параметричному просторі і введено певну метрику, спочатку визначається рангова конфігурація цих станів системи. Після цього застосовується алгоритм для кодування їх DRP-кодом, розроблений в рамках даної роботи.

У випадку, коли введення метрики неможливе (наприклад, для системи розпізнавання образів), автор пропонує використовувати штучну нейронну мережу (ШНМ) в якості кодера.

### **2.3.1 Опис природної нейронної мережі**

Відомо що вперше інтерес до нейромереж був обумовлений піонерською працею МакКалока і Питса, виданою у 1943 році, в якій була запропонована схема комп'ютера, заснованого на аналогії з роботою людського мозку. Вони створили спрощену модель нервової клітини – нейрон [8].

Мозок людини складається із білої і сірої речовин: біла - це тіла нейронів, а сіра - це з'єднувальна тканина між нейронами, або інша назва аксони і дендрити. Мозок складається приблизно з 1011 нейронів, пов'язаних між собою. Кожний нейрон одержує інформацію через свої дендрити, а передає її далі тільки через єдиний аксон, який розгалужується на кінці на тисячі синапсів (рисунок 2.2).

Найпростіший нейрон може мати до 10000 дендритів, що приймають сигнали від інших клітин. Таким чином, мозок містить приблизно 1015 взаємозв'язків. Якщо врахувати, що будь-який нейрофізіологічний процес

активізує відразу множину нейронів, то можна уявити собі ту кількість інформації або сигналів, що виникає в мозку [8].

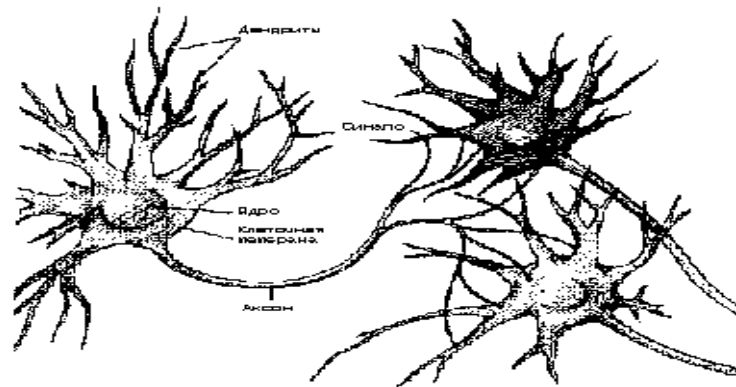


Рисунок 2.6 – Будова природної нейронної мережі

Нейрон взаємодіє за допомогою серії імпульсів, що тривають декілька мілісекунд, кожний імпульс являє собою частотний сигнал з частотою від декілька одиниць до сотень герц. Це дуже повільно в порівнянні із сучасним комп'ютером, але в теж час людський мозок набагато швидше машини може обробляти аналогову інформацію, наприклад: розпізнавання зображення, відчуття смаку, розпізнавання звуків, читання незнайомого почерку, оперування якісними параметрами, тощо. Все це реалізується за допомогою мережі нейронів, з'єднаних між собою синапсами. Іншими словами, мозок - це система з паралельних процесорів, що працюють набагато ефективніше, ніж популярні зараз послідовні обчислення.

### 2.3.2 Архітектура нейронних мереж

Сьогодні нейронні мережі представляють найбільший інтерес в області прийняття рішень. Вони використовують множину простих обчислювальних елементів, що називаються нейронами, кожний з яких імітує поведінку окремої клітини людського мозку. Загальноприйнято вважати, що людський мозок є природною нейронною мережею, тоді як модель мозку може бути представлена як просто нейронна мережа. На рисунку 2.7 показана базова

структура такої нейронної мережі [9].

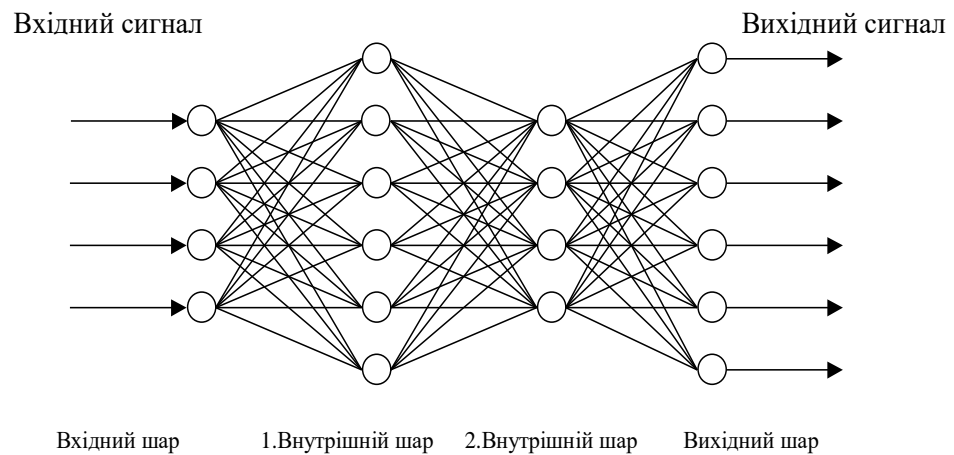


Рисунок 2.7 – Базова структура нейронної мережі

Множина математичних моделей нейрона може бути побудована на базі простої концепції будівлі нейрона. На рисунку 2.8 показана загальна схема математичної моделі нейрона.

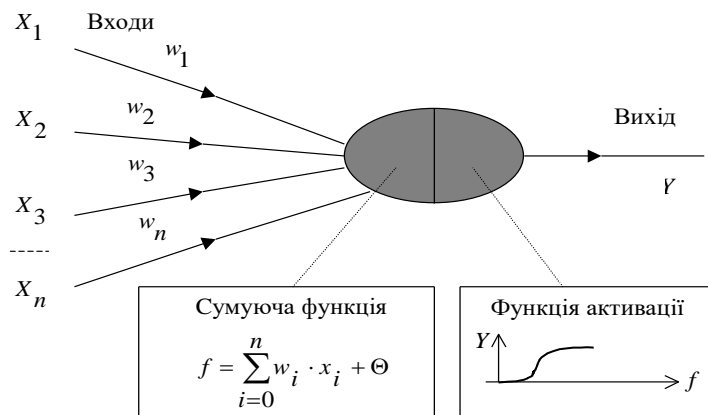


Рисунок 2.8 – Проста математична модель нейрона

Штучним нейроном називається простий елемент, який спочатку обчислює зважену суму  $V$  вхідних величин  $x_i$

$$V = \sum_{i=1}^N W_i \cdot x_i = \vec{W} \cdot \vec{X}, \quad (2.15)$$

де  $N$  - розмірність простору вхідних сигналів.

Так називана функція активації розраховує вихідний сигнал нейрона  $Y$  за рівнем активності  $f$ . Функція активації звичайно є сигмоїдною, як показано в правій нижній рамці на рисунку 2.8.

Потім отримана сума порівнюється з пороговою величиною  $W_0$ , після чого вступає в дію нелінійна функція активації  $f$ . Коефіцієнт  $\{W_i\}$  у зваженій сумі зазвичай називають синаптичним коефіцієнтом або вагою. Сама ж зважена сума  $V$  називається потенціалом нейрона [10].

Як ще один ваговий коефіцієнт при постійному вхідному сигналі можна розглядати величину порогового бар'єру. У цьому випадку ми говоримо про розширений вхідний простір: нейрон із  $N$ -мірним входом має  $N+1$  ваговий коефіцієнт. Якщо ввести у рівняння порогову величину  $W_0$ , то воно переписеться таким чином:

$$V = \sum_{i=1}^N W \cdot x + W_0. \quad (2.16)$$

У залежності від засобу перетворення сигналу і характеру активації виникає різноманітні види нейронних структур. Існують детерміновані нейрони, коли активізуюча функція однозначно обчислює вихід по входу, і ймовірнісні нейрони, стан яких у момент  $t$  є випадкова функція потенціалу і стану у момент  $t-1$ .

### 2.3.3 Функції активації

В штучних нейронах можуть бути різноманітні функції активації, але найпоширенішими є наступні функції:

- лінійна: вихідний сигнал нейрона рівний його потенціалу,
- порогова: нейрон вибирає рішення з двох варіантів: 0 / 1,
- багатопорогова: вихідний сигнал може приймати одне з  $q$  значень, обумовлених  $(q-1)$  порогом всередині граничних значень.

- сигмоїдна: розглядається два види сигмоїдних функцій: з вихідними значеннями в проміжку  $[0,1]$  та з вихідним значенням у проміжку  $[-1,1]$ .

$$s = f(V) = \frac{1}{1 + \exp(-bV)}, \quad (2.17)$$

$$s = f(V) = \frac{\exp(bV) - 1}{\exp(bV) + 1}. \quad (2.18)$$

Коефіцієнт  $b$  визначає крутизну сигмоїда. Оскільки сигмоїдна функція є відображенням  $(-\infty, \infty)$  на  $(-1, 1)$ , то крутизну можна врахувати через величини ваг і порогів, і без обмеження спільності можна прийняти її рівній одиниці [10].

Слід зазначити, що сигмоїдна функція диференціюєма на всій осі абсцис, що використовується в деяких алгоритмах навчання. До того ж, вона має властивість підсилювати слабкі сигнали краще, чим сильні, і запобігає насиченню від сильних сигналів, тому що вони відповідають областям аргументів, де сигмоїд має положистий нахил.

#### 2.3.4 Типи архітектур нейромереж

З точок на площині і з'єднань між ними можна побудувати множину графічних фігур, які називаються графами. Нейронну мережу ми отримаємо у тому випадку, якщо з'єднання між точками уявимо собі як дендрити і синапси, а кожну точку - як один нейрон.

Але не кожне з'єднання нейронів буде працездатним або взагалі доцільно. Тому на сьогоднішній день існує тільки декілька працюючих і реалізованих програмно архітектур нейромереж. Опишемо їх склад і класи вирішуваних ними задач [11].

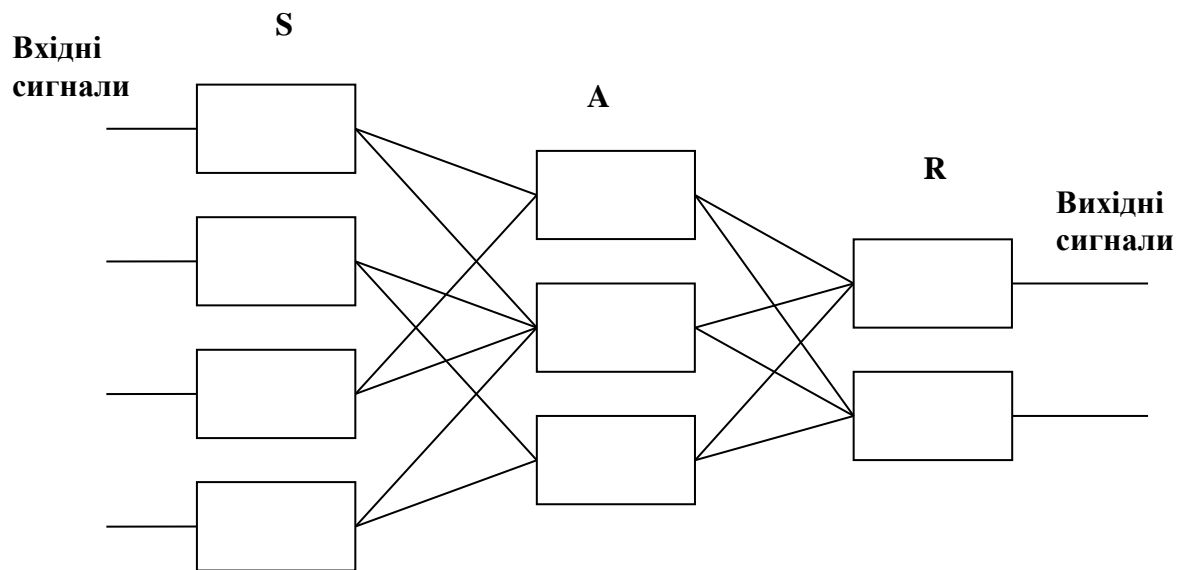


Рисунок 2.9 – Мережі прямого поширення:

S – рецепторний шар; A – аналізуючий шар; R – реагуючий

За архітектурою зв'язків нейромережі можуть бути згруповані у два класи: мережі прямого поширення, у яких зв'язок не має петлі (рисунок 2.9), мережі рекурентного типу, у яких можливий зворотний зв'язок (рисунок 2.10).

Мережі прямого поширення розділяються на одношарові перцептрони (мережі) і багатошарові перцептрони (мережі). Назву перцептрона для нейромережі придумав американський нейрофізіолог Ф. Розенблатт, що придумав в 1957 році перший нейропроцесорний елемент (НПЕ), тобто нейромережу. Він же довів збіжність області рішення для перцептрона при його навчанні.

Відразу після цього почалося інтенсивне дослідження у цій області і було створено перший нейрокомп'ютер Mark I. Відмінність багатошарових мереж полягає в тому, що декілька прихованих прошарків нейронів розташовуються між вхідними і вихідними даними, що додає більше

нелінійних зв'язків у модель.

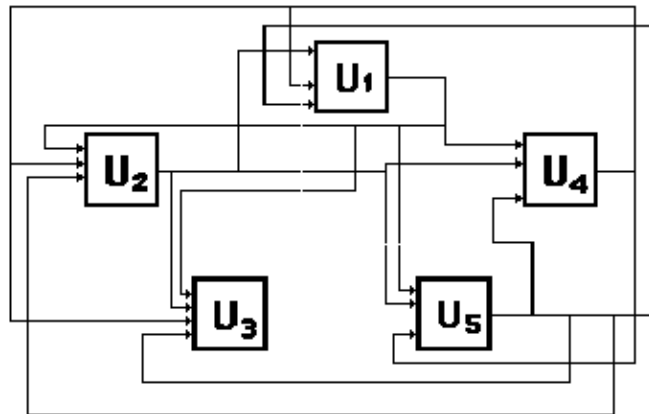


Рисунок 2.10– Рекурентна мережа

Розглянемо структуру пристрою найпростішої багатошарової нейронної мережі. Будь-яка нейронна мережа складається з вхідного та вихідного шарів, де подаються відповідно залежні та незалежні змінні. Вхідні дані піддаються обробці нейронами мережі, а їх вихід порівнюється з очікуваним результатом. У випадку перевищення заданого відхилення вносяться зміни в порогові значення та ваги зв'язків між нейронами. Цей процес повторюється, і якщо відхилення стає меншим за задану похибку, процес навчання завершується. У багатошаровій мережі, крім вхідного та вихідного шарів, існують приховані прошарки. Ці прошарки містять нейрони, які не мають прямих зв'язків із вхідними даними, але взаємодіють лише з виходами вхідного та вхідного прошарків. Таким чином, прихований прошарок додає додаткові перетворення та вносить нелінійності в модель [12].

Якщо одношарова нейромережа дуже добре справляється із задачами класифікації, тому що вихідний прошарок нейронів порівнює отримані від попереднього прошарку значення з порогом і видає значення або нуль, тобто менше порогового значення, або одиницю - більше порогового (для випадку



порогової внутрішньої функції нейрона), і не спроможна вирішувати більшість практичних задач (що було доведено Мінським і Пайпертом), то багатошаровий перцептрон із сигмоїдними вирішувальними функціями спроможний апроксимувати будь-яку функціональну залежність (це було доведено у вигляді теореми). Але при цьому не відомо ні потрібне число прошарків, ні потрібна кількість прихованих нейронів, ні необхідний для навчання мережі час.

Поведінка рекурентних мереж описується диференціальним або різницеvim рівнянням, як правило, першого порядку. Це дуже поширює галузь застосування нейромереж і засоби їх навчання. Організована мережа таким чином, що кожен нейрон отримує вхідну інформацію від іншого нейрона, від навколишнього середовища, або, навіть, від самого себе. Мережі такого типу мають важливе значення, тому що вони дають змогу моделювати нелінійні динамічні системи.

Серед рекурентних мереж можна виділити мережі Хопфілда і мережі Кохонена.

Мережа Хопфілда може обробляти різноманітні зразки, такі як неупорядковані (рукописні літери), впорядковані у часі (тимчасові ряди) або просторові (графіки). Ця рекурентна нейромережа, запропонована Хопфілдом, складається з  $N$  нейронів, які взаємодіють один з одним, за винятком взаємодії з самими собою, і всі нейрони є вихідними. Мережу Хопфілда можна використовувати як асоціативну пам'ять.

Мережа Кохонена, також відома як "карта ознак з самоорганізацією", призначена для самостійного навчання. Під час процесу навчання не обов'язково подавати їй правильні відповіді. Під час навчання мережі подаються різноманітні зразки, і вона вивчає особливості їх структури, розділяючи їх на кластери. Після завершення навчання мережа може класифікувати нові приклади, відносячи їх до одного з кластерів за допомогою критерію "близькості".

Мережа складається з одного вхідного і одного вихідного прошарку.

Кількість елементів у вихідному прошарку безпосередньо визначає, скільки різноманітних кластерів мережа зможе розпізнати. Кожний із вихідних елементів одержує на вхід весь вхідний вектор. Як і у будь-якій нейронній мережі, кожному зв'язку приписана деяка синаптична вага. У більшості випадків кожен вихідний елемент з'єднаний також із сусідніми. Це міжшарові зв'язки відіграють важливу роль в процесі навчання, тому що коригування ваг відбувається тільки біля того елемента, який найкращим чином відкликається на черговий вхід.

### 2.3.5 Навчання багатошарової мережі

Головна відмінність і перевага нейромереж перед класичними засобами прогнозування і класифікації полягає в їх спроможності до навчання.

На етапі навчання відбувається обчислення синаптичних коефіцієнтів в процесі рішення нейронною мережею задач, в яких потрібна відповідь визначається не по правилам, а за допомогою прикладів, згрупованих у навчальні множини. Отже на етапі навчання нейромережа у процесі підготування даних для побудови експертної системи сама виконує роль експерта. Передбачається, що правила знаходяться в структурі навчальних даних [12].

Навчальні дані потрібні для навчання нейронної мережі. Вони повинні відповідати властивостям представлення і випадковості або послідовності. Все залежить від класу задач, що розв'язується. Такі дані являють собою ряд прикладів з вказівкою для кожного з них значення вихідного параметру, що було б бажано отримати. Дії, які відбуваються при цьому процесі, можна назвати контрольованим навчанням: "вчитель" на вихідний вузол повідомляє бажане значення результату обчислень а на вхід мережі подає вектор вихідних даних. Як рішення оптимізаційної задачі можна розглядати контрольоване навчання нейромережі. Її метою є мінімізація функції помилки  $E$  на даній множині прикладів шляхом вибору вагових значень  $W$ . Збіжністю процесу навчання називається досягнення мінімуму. Саме

можливість цього і підвела Розенבלата. Так як від ваг помилка залежить нелінійно, неможливо отримати рішення в аналітичній формі, в результаті чого пошук глобального мінімуму здійснюється за допомогою ітераційного процесу - так званого навчального алгоритму. На даний момент існує вже більше сотні різних навчальних алгоритмів, які відрізняються один від одного критерієм помилки та стратегією оптимізації. Зазвичай в якості міри похибки береться середньо квадратична помилкам (СКП):

$$E = \sqrt{\frac{\sum_{i=1}^M (d_i - y_i)^2}{M}}, \quad (2.19)$$

де  $M$  - кількість прикладів у навчальній множині.

Мінімізація розміру  $E$  здійснюється за допомогою градієнтних методів. Зміна ваг відбувається у напрямку, оберненому до напрямку найбільшої крутизни для функції:

$$W(t+1) = W(t) + \varepsilon \cdot \frac{\partial E}{\partial W}, \quad (2.20)$$

де  $\varepsilon$  - параметр, що визначається користувачем, який називається коефіцієнтом навчання.

Правило навчання Хеба: Збільшувати вагу активного входу нейрона, якщо вихід цього нейрона повинний бути активним.

Зменшити вагу активного входу нейрона, якщо вихід цього нейрона не повинний бути активним.

Це правило, назване правилом Хеба, має найвищий пріоритет перед іншими навчальним правилам, включаючи найбільше використовуваний у даний час метод зворотного поширення помилки (Error Back Propagation Algorithm) [13].

Отже, процедура побудови повного алгоритму навчання нейромережі з

використанням методу зворотного поширення виглядає наступним чином.

1. Подати на входи мережі один із можливих способів і в режимі звичайного функціонування НМ, коли сигнали поширюються від входів до виходів, розрахувати значення останніх. Нагадаємо, що

$$s_j^{(n)} = \sum_{i=0}^M y_i^{(n-1)} \cdot w_{ij}^{(n)}, \quad (2.21)$$

де  $M$  - число нейронів у шарі  $n-1$  з врахуванням нейрона з постійним вихідним станом  $+1$ , що задає зсув;  $y_i^{(n-1)} = x_{ij}^{(n)}$  -  $i$ -ий вхід нейрона  $j$  шару  $n$ .

$$y_j^{(n)} = f(s_j^{(n)}), \quad (2.22)$$

$$y_q^0 = I_q, \quad (2.23)$$

де  $I_q$  -  $q$ -а компонента вектора вхідного образу.

2. Розрахувати  $\delta^{(N)}$  для вихідного шару по формулі. Розрахувати зміни ваг  $\Delta w^{(N)}$  шару  $N$ .

3. Розрахувати відповідно  $\delta^{(n)}$  і  $\Delta w^{(n)}$  для всіх інших шарів,  $n = N-1, \dots, 1$ .

4. Скорегувати усі ваги в НМ:

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t). \quad (2.24)$$

5. Якщо помилка мережі істотна, перейти на крок 1. У протилежному випадку - кінець.

Мережі на кроку 1 поперемінно у випадковому порядку пред'являються всі тренувальні способи, щоб мережа, образно говорячи, не забувала попередні в міру запам'ятовування інших.

## 2.4 Використання нейронної мережі в якості кодера

Розглянемо алгоритм навчання на простій модельній задачі.

Навчання перцептрона проходить наступним чином. Подається сукупність (множина) зображень по одному на вхід перцептрона та змінюються ваги до того моменту, доки для всіх зображень не буде досягнуто необхідний вихід. Припустимо, що вхідне зображення відображається на демонстраційній картці. Усі карти розділено на квадрати і подається вхідний сигнал на перцептрон від кожного квадрата. Якщо в квадраті немає лінії, то від неї подається нуль, у протилежному випадку – одиниця. Сукупність одиниць і нулів, котрі подаються на входи перцептрона задається сукупністю квадратів на картці. Основною метою є навчити перцептрон, за умови подавання на нього сукупності входів, що задають непарне число, вмикати індикатор, і у випадку парного числа не робити цього [15].

На рисунку 2.11 показана така перцептронна конфігурація.

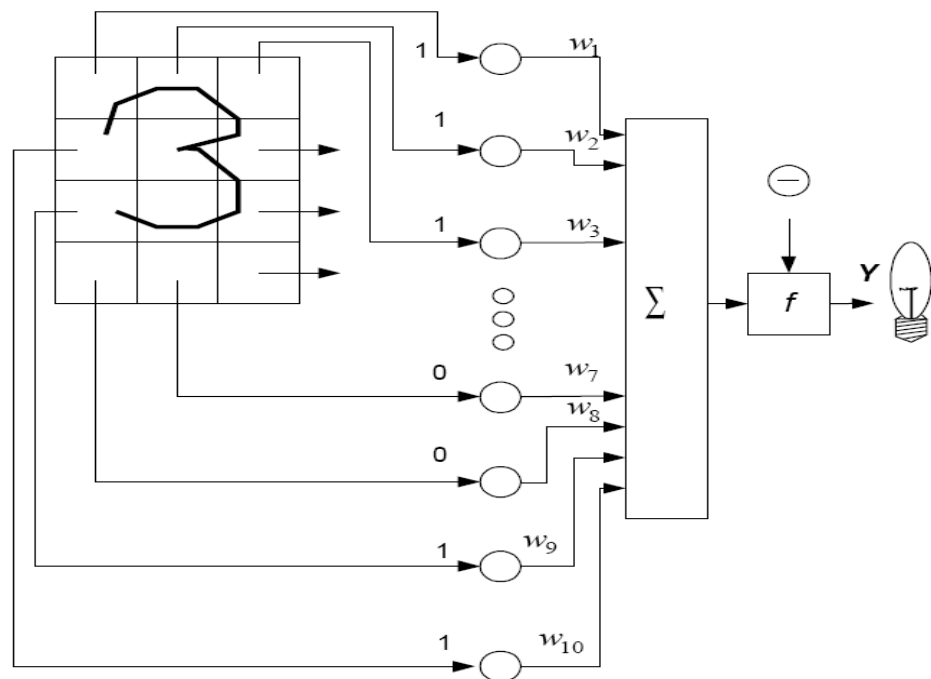


Рисунок 2.11 – Принцип побудови перцептронної системи розпізнавання

Припустимо, що вектор  $x$  є зображенням розпізнаваної демонстраційної карти. Кожну компоненту  $x_i$  (квадрат зображення карти) вектора  $x$  перемножують на відповідну компоненту  $w_i$  вектора ваг  $w$ . Ці добутки сумують. У випадку якщо сума не перевищує поріг  $q$ , то вихід нейрона у дорівнює нулю, у протилежному випадку – одиниця (індикатор запалюється). Ця операція стисло записана у векторній формі:  $y = xw$ . Щоб навчити мережу, зображення  $x$  дається як вхід, а вихід  $y$  обчислюється. Нічого не змінюється у разі якщо вихід правильний. В протилежному випадку щоб зменшити помилку, приєднані до входів що підсилюють помилковий результат, ваги, модифікуються.

Перевірка адекватності запропонованої математичної моделі ШНМ як кодера рангової конфігурації вимагає синтезу нетипової нейромережі, яка, у першому наближенні, матиме одношарову неповнозв'язну архітектуру. Наявність зв'язків між нейронами мережі визначатиметься матрицею інцидентності модельованого графа, а матриця ваг міжнейронних зв'язків – матрицю суміжності для рангів модельованого графа. Найближчою за архітектурою відомою нейромережею є нейромережа Хопфілда. Надходження на вхід такої мережі деякого набору ознак переводить нейромережу в одне з положень рівноваги, що дозволяє асоціювати вхідний вектор ознак із одним із об'єктів, на розпізнавання яких мережу навчено. Структурну схему такої нейромережі зображено на рисунку 2.12.

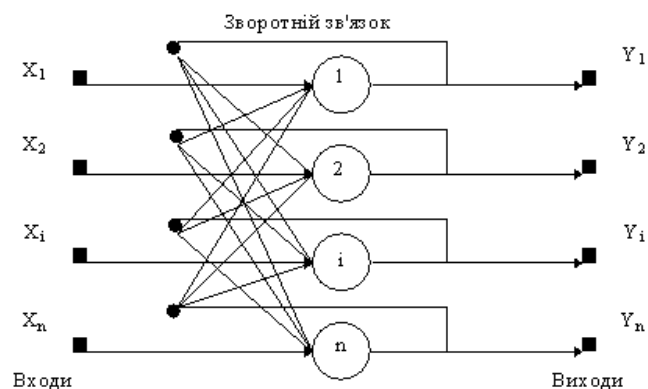


Рисунок 2.12 - Архітектура нейромережі Хопфілда

Дослідження запропонованої моделі нейронної мережі у вигляді рангової конфігурації полягало у вирішенні класичної задачі розпізнавання цифр від 0 до 9. Зображення кожної із цифр закодовано у вигляді поля  $256 \times 256$  пікселів. За умови, що виходу кожного нейрону створеної нейромережі відповідає цифра від 0 до 9 безпосередньо (вектор  $Y$ ), кількість нейронів в мережі дорівнює десяти. На вхід нейромережі подавався вектор  $X$  із десяти ознак, кожен елемент якого дорівнював  $+1$  або  $-1$  в залежності від того, яких за значенням пікселів більше у кожній з 10 ділянок розміром  $40 \times 40$  пікселів, розташованих на зображенні за вибраною схемою, наведеною на рисунку 2.13.

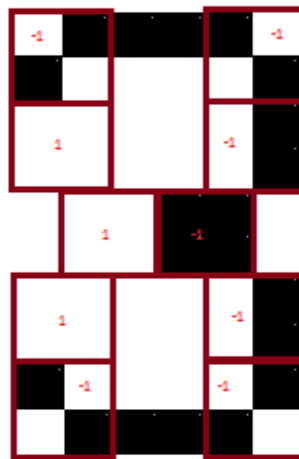


Рисунок 2.13 – Схема розташування площинки-ознак на рисунку цифри 3

Відповідно запропонованої математичної моделі, матрицею ваг нейромережі буде трикутна матриця із нульовою діагоналлю, під якою розташовуватимуться ранги міжнейронних зв'язків. Приклад матриці наведено на рисунку 2.14. Такий вигляд матриці ваг зумовлено потребою виключення зворотних та циклічних зв'язків, не передбачених матрицею інцидентності запропонованої математичної моделі.

Для проведення експерименту було обрано середовище Matlab, яке є одним з існуючих систем, які дозволяють в досить простій формі реалізувати нейронні мережі.

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	26	0	0	0	0	0	0	0	0	0	0	0
3	44	36	0	0	0	0	0	0	0	0	0	0
4	3	35	36	0	0	0	0	0	0	0	0	0
5	33	16	7	17	0	0	0	0	0	0	0	0
6	41	11	19	28	31	0	0	0	0	0	0	0
7	9	32	43	30	18	34	0	0	0	0	0	0
8	41	19	9	10	11	44	9	0	0	0	0	0
9	37	43	13	26	26	20	4	42	0	0	0	0
10	30	30	3	44	6	14	17	36	22	0	0	0
11	30	5	7	40	13	39	45	15	35	21	0	0
12	25	36	4	44	4	18	13	28	22	12	36	0
13	7	32	33	29	18	34	11	7	15	9	35	0
14	28	5	4	19	40	29	7	1	18	32	45	0
15	17	10	24	20	28	39	41	19	32	6	43	0
16	20	43	36	1	17	18	20	41	41	7	33	0
17	27	10	4	16	30	45	26	8	34	32	44	0
18	40	2	30	35	28	15	1	14	7	19	6	0

Рисунок 2.14 – Приклад матриці рангів ваг міжнейронних зв'язків

$\text{Net} = \text{NEWHOP}(T)$  – функція для створення рекурентної мережі Хопфілда, використовує один вхідний аргумент, де  $T$  – матриця розмірністю  $R \times Q$ , при цьому  $Q$  – цільовий вектор (значення повинні бути +1 або -1), повертає нову рекурентну нейронну мережу Хопфілда зі стабільними точками в векторі  $T$ ;  $R$  – число елементів вектору входу.

$[Y, Pf, Af] = \text{sim}(\text{net}, 10, [], X)$  – моделювання нейронної мережі.  $Y$  – виходи мережі.

Коли мережа Хопфілда спроектована, її можна перевірити з одним або великим числом векторів входу. Вхідні вектори, близькі до точки рівноваги цілі, швидше за все, знайдуть ціль. Перевірити мережу за відносно короткий час дозволяє спроможність мережі Хопфілда досить швидко обробляти набори векторів входу.

Після запуску створеної нейромережі отримуємо такі результати: на рисунку 2.15 наведено зображення еталонних цифр, які були використанні для навчання.

Отже, результати експерименту показали, що створена, на основі запропонованої математичної моделі, нейромережа якісно виконує задачу розпізнавання цифр, що підтверджує адекватність математичної моделі представлення нейронної мережі ранговими конфігураціями.



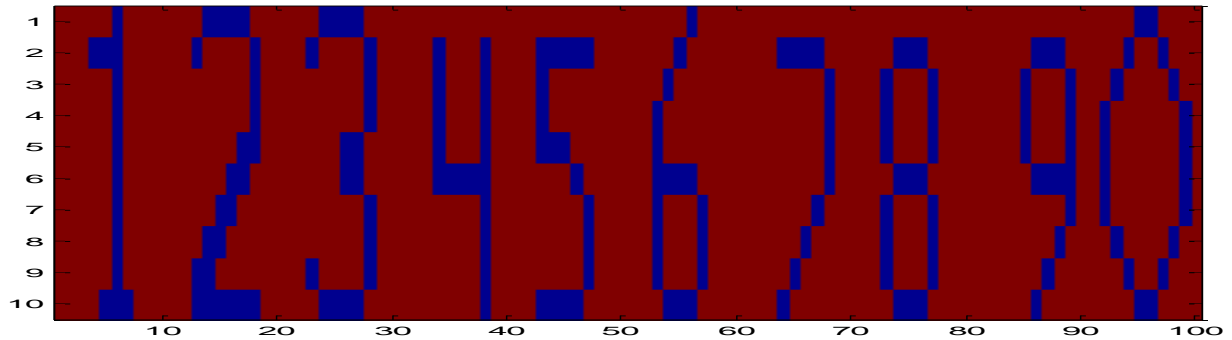


Рисунок 2.15 – Еталонні зображення цифр

## 2.5 Дослідження коригуючих властивостей рангового коду

### 2.5.1 Експериментальні дослідження

В теорії передачі інформації по каналах зв'язку важливе місце займає надійність передачі, яка визначається властивістю самого каналу передачі інформації з однієї сторони, а з іншої коригуючими властивостями коду, що використовується для передачі інформації. В розділі 2.1 даної роботи було показано, що вірогідність невиявлення помилки для запропонованого DRP-коду на декілька порядків менша від такої ж величини стандартного КПВ коду МТА-3 при статистичній ймовірності внесення помилки каналом з завадами. Оскільки канали зв'язку в інформаційних мережах систем управління працюють в умовах промислового виробництва, в якому можливі значні електромагнітні завади, то вони можуть вносити помилки в кодові слова, що передаються мережею. Тому однією з основних задач в цьому випадку є визначення коригуючих властивостей запропонованого рангового коду і розробка алгоритмів локалізації і виправлення помилок в кодових словах, внесених завадами каналу.

Як вказано у розділі 2.1, представлений DRP-код є кодом з постійною вагою (КПВ) за своєю природою. Для масиву з  $m$  кодованих елементів він складається з  $m$  кодових слів, при цьому кожне кодове слово має  $m-1$

одиниць. Згідно зі стандартною класифікацією, КПВ відноситься до блокових нелінійних кодів. Аналіз математичних моделей рангового коду вказує на те, що це є множина  $m$  багатозначних логічних функцій значень  $n$ . Значення аргументів отримуються відображенням впорядкованих відстаней між елементами вихідного простору на впорядковану множину рангів за допомогою логічної операції  $I$ . Оскільки операція  $I$  над набором кодових слів не є групою, стандартні методи для виправлення помилок передачі коду DRP не застосовуються. Це вимагає розробки унікальних алгоритмів локалізації та виправлення помилок, які базуються на вивченні евристичних шаблонів. Оскільки асиметричні помилки в кодових словах можуть виявлятися стандартними алгоритмами підрахунку кількості одиниць у кодовому слові, основна увага в даному випадку зосереджена на дослідженні симетричних помилок, таких як попарна заміна окремої цифри кодового слова з "0" на "1" і навпаки.

Для дослідження виберемо рангову конфігурацію трьохмірного симплексу, зображену на рис. 2.1 а, б. Матриця інцидентностей (яка представляє DRP-код) і матриця суміжностей цієї конфігурації представлені на рис.2.2 в розділі 2.1.

Дослідження буде полягати в аналізі вигляду DRP-коду після внесення в вибране кодове слово парної помилки. Кількість різних типів помилок  $E_v$  можна визначити за такою формулою:

$$E_v = m(C_{m-1}^1 C_{(n-m+1)}^1), \quad (2.25)$$

де  $m$  – кількість кодованих символів (станів системи),  $C_{m-1}^1$  - кількість можливих помилок в одиничних розрядах,  $C_{(n-m+1)}^1$  - кількість можливих варіантів помилок в нульових розрядах. Для DRP-коду вибраної для дослідження конфігурації (рис. 2.1 і рис. 2.2,  $m=4$ ,  $n=6$ ) кількість варіантів помилок для дослідження дорівнює:

$$E_v = 4(C_{4-1}^1 C_{(6-4+1)}^1) = 4 \cdot 3 \cdot 3 = 36.$$

Аналіз результатів проведеного дослідження для цих варіантів показав, що для побудови евристичного алгоритму, окрім самих помилкових кодових слів потрібно аналізувати також перевірочну матрицю  $G_R[n,n]$  рангових відстаней між всіма парами кодових слів. Приклади правильного *DRP*-коду і перевірочної матриці для нього і помилкових *DRP*- кодів і перевірочних матриць для них наведені на рис. 2.16 – 2.18.

Ранги → Символи ↓	6	5	4	3	2	1
A	1	0	1	0	0	1
B	0	1	1	0	1	0
C	1	0	0	1	1	0
D	0	1	0	1	0	1

а)

Ранги → Відстані ↓	6	5	4	3	2	1
d(AC)	1	0	0	0	0	0
d(AB)	0	0	1	0	0	0
d(AD)	0	0	0	0	0	1
d(BC)	0	0	0	0	1	0
d(BD)	0	1	0	0	0	0
d(CD)	0	0	0	1	0	0

б)

Рисунок 2.16 – Правильний *DRP*-код рангової конфігурації (а) і перевірочна матриця рангових відстаней для нього (б)

Ранги → Символи ↓	6	5	4	3	2	1
A	1	0	1	0	0	1
B	1	0	1	0	1	1
C	1	0	0	1	1	0
D	0	1	0	1	0	1

а)

Ранги → Відстані ↓	6	5	4	3	2	1
d(AC)	1	0	0	0	0	0
d(AB)	1	0	1	0	0	0
d(AD)	0	0	0	0	0	1
d(BC)	1	0	0	0	1	0
d(BD)	0	0	0	0	0	0
d(CD)	0	0	0	1	0	0

б)

Рисунок 2.17 –DRP-код рангової конфігурації (а) з помилкою в кодовому слові В (“1” з розряду 5 перейшла в 6 розряд, а на її місці став “0”) і перевірна матриця рангових відстаней для нього (б)

Ранги → Символи ↓	6	5	4	3	2	1
A	1	0	1	0	0	1
B	0	0	1	0	1	1
C	1	0	0	1	1	0
D	0	1	0	1	0	1

а)

Ранги → Відстані ↓	6	5	4	3	2	1
d(AC)	1	0	0	0	0	0
d(AB)	0	0	1	0	0	1
d(AD)	0	0	0	0	0	1
d(BC)	0	0	0	0	1	0
d(BD)	0	0	0	0	0	1
d(CD)	0	0	0	1	0	0

б)

Рисунок 2.18 –DRP-код рангової конфігурації (а) з помилкою в кодовому слові В (“1” з розряду 5 перейшла в 1 розряд, а на її місці став “0”) і перевірна матриця рангових відстаней для нього (б)

## 2.5.2 Теоретична оцінка коригувальної здатності рангових кодів

Результати проведених у попередніх розділах досліджень вказують на те, що, відмінно від відомих стандартних кодів з постійною вагою (КПВ), які лише виявляють наявність помилок, але не мають здатності їх локалізувати та виправляти, запропоновані рангові коди дозволяють виявляти, локалізувати і виправляти відносно меншу кількість симетричних одиночних помилок. Оскільки цей результат отримано для рангової конфігурації розміром  $m=4$ , виникає завдання теоретичної оцінки локалізуючої та коригуючої здатності рангових кодів для рангових конфігурацій різної розмірності.

Оскільки канали передачі в промислових мережах, де планується використовувати запропонований ранговий код, піддаються впливу завад, які можуть породжувати як симетричні, так і асиметричні помилки, використання методів локалізації та коригування помилок, що застосовуються для стандартних КПВ кодів, стає неможливим. Таким чином, теоретичну оцінку ефективності цих методів слід проводити, враховуючи принципи конструкції рангових кодів. Відповідно до характеристик рангових кодів [3], при збільшенні кількості кодованих станів системи  $m$ , розрядність  $n$  коду зростає за виразом:

$$n = \frac{m(m-1)}{2}, \quad (2.26)$$

а кількість одиниць в кодовому слові теж збільшується і дорівнює  $m-1$ .

Для отримання теоретичної оцінки доведемо спочатку теорему: для рангової конфігурації розмірності  $m$  відстань Хемінга  $d_x$  між словами рангового коду дорівнює:

$$d_x = 2(m-1) - 2. \quad (2.27)$$

Доведення: У двох кодових словах рангового коду, між якими визначається відстань Хемінга, міститься  $2(m-1)$  одиничних бітів, при цьому ці одиничні біти співпадають лише в одному позначенні, по одному відповідному біту у кожному слові, загалом 2 співпадіння. Таким чином, кількість бітів, де одиниці не збігаються, складає  $2(m-1)-2$ , і логічна операція XOR над цими словами, яка визначає відстань Хемінга, дає  $2(m-1)-2$  одиниці, як і потрібно було довести.

З отриманої теореми, відповідно до [23], можна зробити висновок, що теоретична оцінка можливості виявлення помилок, кратністю  $g_d$  визначається за формулою

$$g_d \leq d_x - 1 = 2(m-1) - 3, \quad (2.28)$$

а коригування помилок кратністю  $g_c$  – за формулою:

$$g_c \leq \frac{(d_x - 1)}{2} = \frac{(2(m-1) - 3)}{2}. \quad (2.29)$$

Адекватність отриманої моделі оцінки коригувальної здатності запропонованих кодів перевірена експериментально на прикладах конкретних рангових конфігурацій розмірності  $m=4$  і  $m=5$  з використанням моделей рангових конфігурацій у вигляді матриць інцидентності, стрічки яких і утворюють потенціальні коди вершин рангового симплекса (рис. 2.19 і 2.20)

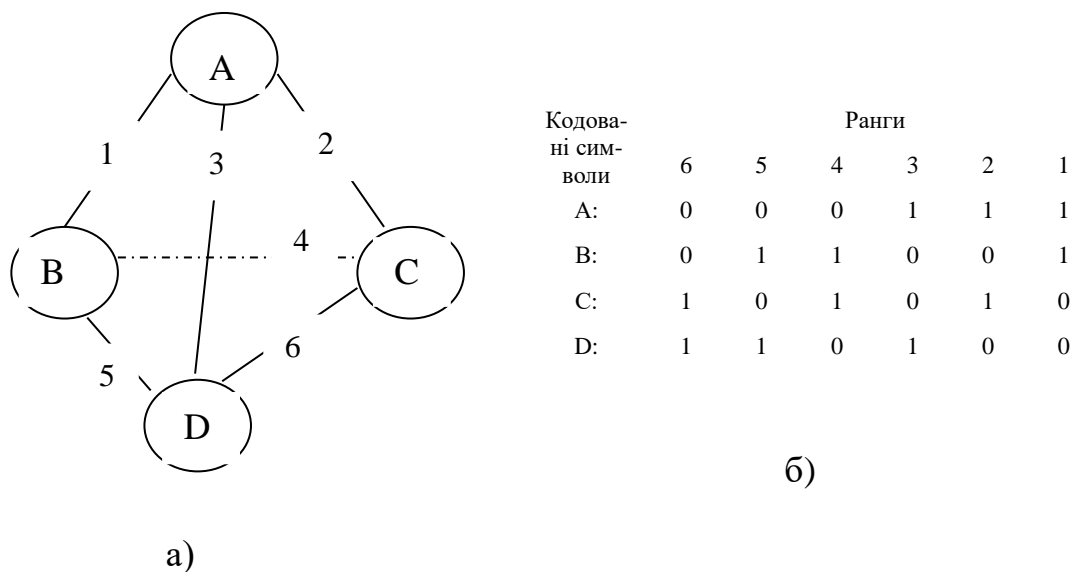


Рисунок 2.19 – Модель рангової конфігурації для  $m=4$ : а) – ранговий симплекс; б) – матриця інцидентності

Теоретичне значення відстані Хемінга для даної моделі згідно формул (2.26) і (2.27) дорівнює:

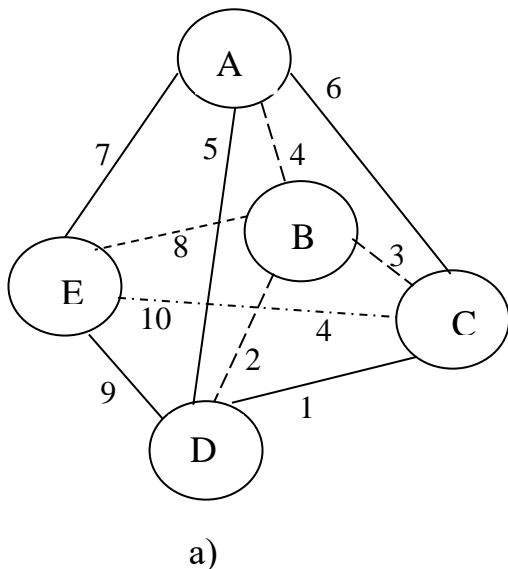
$$d_x = 2(4-1) - 2 = 4.$$

Обрахувавши відстані між всіма кодовими словами DRP-коду, записаними в матриці інциденцій, отримаємо:

$$\begin{aligned} d_x(A,B) &= 000111 \oplus 011001 = 4 & d_x(A,C) &= 000111 \oplus 101010 = 4 \\ d_x(A,D) &= 000111 \oplus 110100 = 4 & d_x(B,C) &= 011001 \oplus 101010 = 4 \\ d_x(B,D) &= 011001 \oplus 110100 = 4 & d_x(C,D) &= 101010 \oplus 110100 = 4 \end{aligned}$$

За формулами (2.28) і (2.29) обрахуємо детектувальну і коригувальну здатність коду відповідно для даної рангової конфігурації:

$$\begin{aligned} g_d &\leq d_x - 1 = 4 - 1 = 3; \\ g_c &\leq \frac{(d_x - 1)}{2} = \frac{3}{2} = 1. \end{aligned}$$



Кодові символи	Ранги									
	10	9	8	7	6	5	4	3	2	1
A:	0	0	0	1	1	1	1	0	0	0
B:	0	0	1	0	0	0	1	1	1	0
C:	1	0	0	0	1	0	0	1	0	1
D:	0	1	0	0	0	1	0	0	1	1
E:	1	1	1	1	0	0	0	0	0	0

б)

Рисунок 2.20 – Модель рангової конфігурації для  $m=5$ : а) – ранговий симплекс; б) – матриця інцидентності

Аналогічні розрахунки виконаємо для рангової конфігурації розмірності  $m=5$ , отримаємо:

$$d_x = 2(5-1) - 2 = 6$$

$$\begin{aligned} d_x(A,B) &= 0001111000 \oplus 0010001110 = 6 & d_x(A,C) &= 0001111000 \oplus 1000100101 = 6 \\ d_x(A,D) &= 0001111000 \oplus 0100010011 = 6 & d_x(A,E) &= 0001111000 \oplus 1111000000 = 6 \\ d_x(B,C) &= 0010001110 \oplus 1000100101 = 6 & d_x(B,D) &= 0010001110 \oplus 0100010011 = 6 \\ d_x(B,E) &= 0010001110 \oplus 1111000000 = 6 & d_x(C,D) &= 1000100101 \oplus 0100010011 = 6 \\ d_x(C,E) &= 1000100101 \oplus 1111000000 = 6 & d_x(D,E) &= 0100010011 \oplus 1111000000 = 6 \end{aligned}$$

Звідси детектувальна здатність коду для рангової конфігурації розмірності  $m=5$  дорівнює  $g_d \leq d_x - 1 = 6 - 1 = 5$ , а коригувальна:

$$g_c \leq \frac{(d_x - 1)}{2} = \frac{5}{2} = 2.$$

За законом індукції легко показати чисельні значення кратності виявлених і коригованих помилок для рангових конфігурацій вищих розмірностей  $m=6..11$ . Для конфігурацій з розмірністю, вищою за 11, потенціальні коди використовувати неефективно, оскільки їх розрядність перевищує розрядність сучасних ПК  $n \geq 64$ , і обчислювальні операції над кодами стають складними і неефективними.

## 2.6 Висновки до розділу

В даному розділі наведено математичне обґрунтування опису ранговими конфігураціями інформації про стани об'єктів систем управління. Математичні моделі рангових конфігурацій використані для визначення характеристик рангових кодів, якими кодують ці конфігурації.

Розроблено структурну схему пристрою передавання рангової інформації в комунікаційних мережах систем автоматизованого управління, виконано математичне обґрунтування роботи кодера інформації у вигляді



штучної нейронної мережі. Наведено результати дослідження роботи такого кодера на прикладі розпізнавання цифр нейронною мережею, як однією з відомих практичних систем прийняття рішень. Результати дослідження показали адекватність представлення нейронної мережі ранговими конфігураціями.

Проведено експериментальні та теоретичні дослідження коригувальної здатності запропонованого рангового коду до локалізації і виправлення помилок, внесених в каналах зв'язку комунікаційних мереж, які показали можливість локалізації і виправлення симетричних помилок в кодових словах і сформулювати вербальний алгоритм для її реалізації.

### 3 РОЗРОБКА АЛГОРИТМІВ І ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

#### 3.1 Розробка алгоритму кодування рангових конфігурацій *DRP*-КОДОМ

Першим кроком в рішенні задачі побудови оптимального RP-коду, що відображає простір кодів заданої конфігурації, є визначення його розрядності. Під оптимальним розуміється код, що має мінімальну з всіх можливих розрядність. Очевидно, що розрядність RP-коду залежить як від числа кодованих елементів, так і від їх заданої просторової конфігурації:  $n = (m, K)$ . Як було показано в розділі 2.1, розрядність DRP-коду визначається виразом

$$n_{\max} = (m(m-1))/2, \quad (3.1)$$

Побудова самого коду, тобто знаходження множини двійкових кодових слів  $B = \{B_1, B_2, \dots, B_m\}$ , конфігурація яких в двійковому просторі ізоморфна заданій конфігурації  $K$  вихідного простору  $m$  станів, виконується за допомогою розробленого в роботі алгоритму.

Для кращого розуміння розроблюваного алгоритму наведемо приклад конкретної рангової конфігурації (рис. 3.1) і наведемо позначення змінних, що використовуються для її опису.

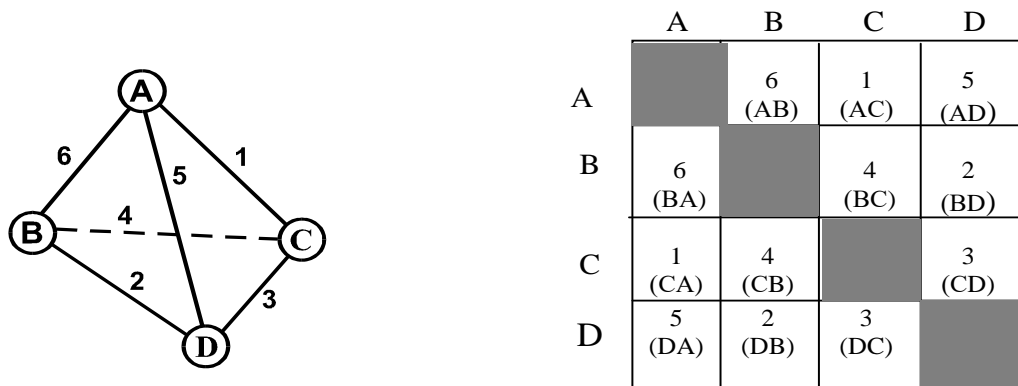


Рисунок 3.1 – Приклад кодованої рангової конфігурації: а – трьохмірний симплекс, б – матриця суміжностей рангів

Словесний опис алгоритму побудови DRP-коду для заданої конфігурації має наступний вигляд:

1. Введемо початкові дані - число  $M$  об'єктів в конфігурації і матрицю інцидентності  $MINC(M, M-1)$ .
2. Обчислюємо розрядність RP-коду за формулою (2.4):

$$N = \frac{M(M-1)}{2}.$$

3. Будуємо початкову кодову матрицю MKOD шляхом виділення в пам'яті масиву MKOD( $M, N$ ) і запису нулів на місцях всіх його елементів. В цій матриці  $i$ -та стрічка відповідає  $i$ -тому кодованому елементу, а  $j$ -ий стовпчик ( $N-j+1$ )-му рангу.

4. Для кожної стрічки кодової матриці MKOD виконуємо процедуру заповнення одиницями DRP-коду наступним чином: переглядаємо всі  $(m-1)$  стовпчиків  $i$ -ої стрічки матриці інцидентностей MINC, і записуємо "1" в стовпчик матриці з номером  $N - MINC(i, j) + 1$ .

5. Виводимо на друк DRP-коди об'єктів, при цьому код  $i$ -го об'єкта відповідає  $i$ -ій стрічці матриці MKOD( $M, N$ ).

Блок-схема алгоритму зображена на рисунку 3.2.

Для заданої на рис.3.1 конфігурації  $n = 4*3/2 = 6$ , а матриця суміжностей рангів  $MINC[4,3]$  має вигляд:

$$MINC = \begin{matrix} A \\ B \\ C \\ D \end{matrix} \begin{matrix} \parallel 6, & 1, & 5 \\ \parallel 6, & 4, & 2 \\ \parallel 1, & 4, & 3 \\ \parallel 5, & 2, & 3 \end{matrix}$$

Коди для кожної вершини матимуть вигляд:

A:	1	1	0	0	0	1
B:	1	0	1	0	1	0
C:	0	0	1	1	0	1
D:	0	1	0	1	1	0

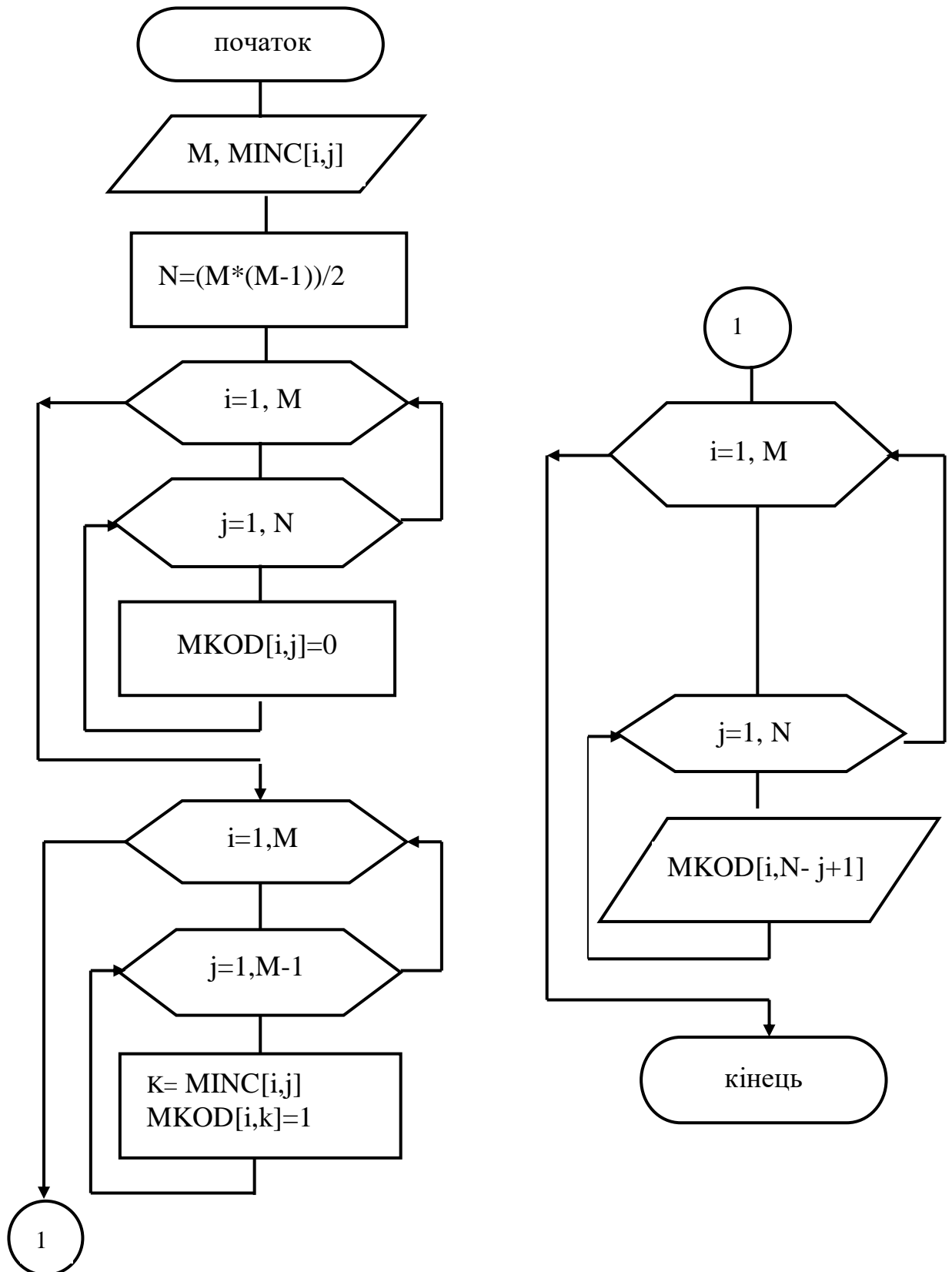


Рисунок 3.2 – Алгоритм кодування рангової конфігурації DRP-кодом

### 3.2 Розробка алгоритму виправлення помилок в *DRP*-коді

Згідно найдених в розділі 2.5 закономірностей евристичний алгоритм локалізації і виправлення поодиноких асиметричних і симетричних помилок в *DRP*-коді можна має наступний вербальний опис.

На 1-му кроці в пам'яті декодера формується двовірний масив розміром  $m \times n$  прийнятих з каналу  $m$  кодових слів, де  $n$  - розрядність коду. За умови безпомилкової передачі цей масив буде відповідати матриці інцидентів переданої рангової конфігурації станів системи, а кожна стрічка масиву - правильному слову *DRP*-коду.

На 2-му кроці шляхом підрахунку одиниць в кожному рядку масиву локалізуються кодові слова з асиметричними помилками у випадку  $t \neq m-1$ , де  $t$  - кількість одиниць в слові. Виправлення асиметричних помилок здійснюється за процедурою: якщо в деякому слові кількість одиниць  $t > m$ , то скануються стовпчики масиву кодових слів і в знайденому помилковому слові замінюються одиниці на нулі в тих розрядах, стовпці яких містять більше 2-х одиниць; якщо в деякому слові кількість одиниць  $t < m-1$ , то скануються стовпчики масиву кодових слів і в знайденому помилковому слові замінюються нулі на одиниці в тих розрядах, стовпці яких містять менше 2-х одиниць.

На 3-ому кроці скануються стовпчики масиву і виявляється наявність симетричних парних помилок у випадку наявності стовпчиків з одиницями, в яких кількість одиниць не дорівнює 2. Локалізація помилкового кодового слова здійснюється з допомогою перевірконої матриці  $G_R[n, n]$  за наступною процедурою: відзначаються ті рядки матриці, в яких міститься два одиничних біти. Якщо таких кодових слів більше одного, то за помилкове вибирається те, індекс якого міститься в індексах пар відстаней двічі.

На 4-ому кроці локалізуються номери пошкоджених бітів і коректуються. Для асиметричних помилок вони визначаються наявністю зайвих чи відсутністю потрібних одиниць в словах, а для симетричних

помилку за правилом: стовпчик з нулями в перевірочній матриці  $G_R[n,n]$  відстаней кодових слів, отриманої за логічною операцією “Г” над всіма парами кодових слів, показує біт з заміною  $1 \rightarrow 0$ , а стовпчик з трьома одиницями - біт з заміною  $0 \rightarrow 1$ .

Схема алгоритму локалізації і виправлення помилок *DRP*-коду наведена в графічній частині магістерської роботи.

### 3.3 Розробка алгоритму декодування *DRP*-коду

На приймальному кінці розробленої системи передачі інформації, після локалізації і виправлення помилок *DRP*-коду згідно з розробленим в розділі 3.2 алгоритмом, необхідно декодувати ранговий код в рангову конфігурацію, тобто виконати процедуру, обернену до процедури, описаної в розділі 3.1. Алгоритм декодування для наглядності будемо описувати для прикладу *DRP*-коду і рангової конфігурації, наведених у вказаному розділі.

Словесний опис алгоритму декодування *DRP*-коду має такий вигляд:

1. Введемо початкові дані – кількість кодованих елементів  $M$  і матрицю  $MKOD(M,N)$ , сформовану з прийнятих з каналу передачі кодових слів.

2. Будуємо початкову матрицю  $MINC$  рангової конфігурації шляхом виділення в пам'яті масиву  $MINC(M,M-1)$  і запису нулів на місцях всіх його елементів. В цій матриці  $i$ -тий рядок відповідає  $i$ -тому кодованому елементу, а  $j$ -ий стовпчик відповідає одному із суміжних даному елементу (символу) рангів.

3. Для кожного рядка матриці суміжних рангів  $MINC(M,M-1)$  виконуємо процедуру заповнення величинами суміжних рангів наступним чином: для вибраного чергового номера рядка матриці  $MINC$  задаємо початковий номер стовпчика  $p=0$  і переглядаємо по черзі всі  $N$  стовпчиків  $i$ -ої стрічки кодової матриці  $MKOD(M,N)$ . При виявленні в деякому стовпчику з номером  $j$  одиничного біта "1" записуємо величину рангу  $N - j + 1$  в  $(p+1)$ -ий стовпчик матриці  $MINC$ .

4. Виводимо на друк матрицю конфігурації  $\text{MINC}(M, M-1)$ .

Блок-схема алгоритму зображена на рисунку 3.3.

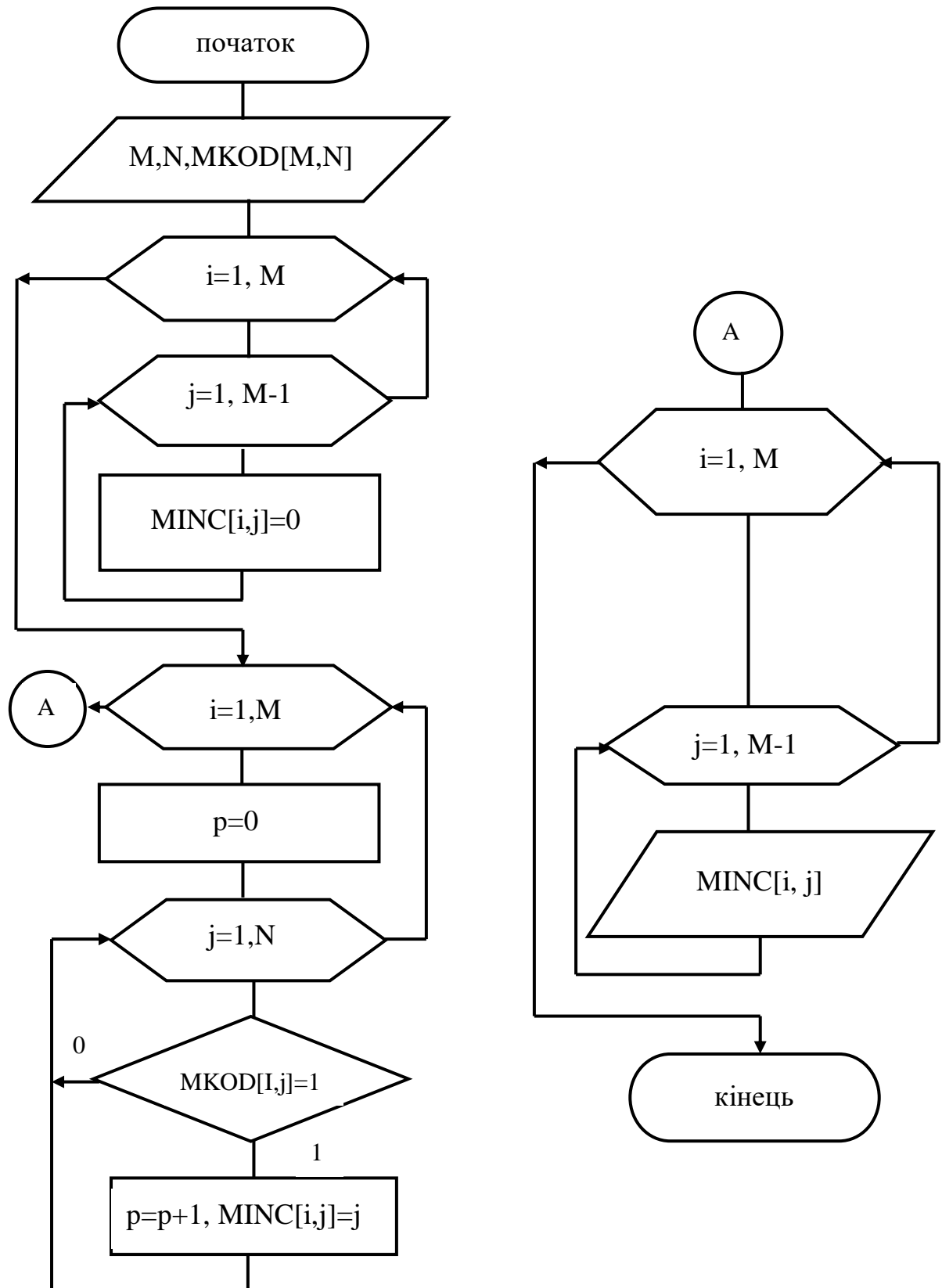


Рисунок 3.3 – Схема алгоритму роботи декодера

### 3.4 Розробка програмного забезпечення

#### 3.4.1 Вибір програмної платформи

Для створення програми, яка буде обчислювати коди рангових конфігурацій обираємо середовище Turbo C оскільки в ньому зручно і легко реалізувати програмну інтерпретацію роботи з масивами чисел.

Середовище дає змогу для програмування використати основні базові конструкції мови C++, такі як: змінні і константи; типи даних; вирази і оператори; функції локальні та глобальні змінні;

#### 3.4.2 Опис програми кодування потенціальним кодом

Вхідними даними є:

- а) кількість об'єктів, які необхідно кодувати;
- б) кількість ознак, які описують дані об'єкти;
- в) матриця суміжностей рангів цього графа.

г) елементи матриці суміжності, тобто безпосередньо ранги ваг нейронної мережі або відстаней між станами системи у просторі параметрів..

При виконанні програми виконується попереднє обнулення матриці, а потім заповнення її рангами. Після цього за збільшенням кожному значенню у відповідність ставиться ранг (для конкретного випадку – кількість рангів дорівнює 21). Звідси буде отримано і код розрядності якого відповідає 21, причому очевидно, що в кожному коді присутні тільки 6 “одиниць”, а всі інші “нули”, оскільки кожний об'єкт зв'язаний з іншими сімома і цьому зв'язку відповідає 1 ранг.

Вихідними даними є:

Матриця інцидентності графа, тобто матриця кодів за заданою конфігурацією.

Лістинг програми на мові Сі наведено в Додатку Б.



### 3.5 Результати тестування програми

Результати розрахунків, які були отримані в результаті машинного експерименту:

Матриця суміжності рангів графа подається на вхід програми як початкові данні для подальшого обчислення.

Матриця суміжності рангів графа:

```

0 1 2 3 4 5 6
1 0 7 8 9 10 11
2 7 0 12 13 14 15
3 8 12 0 16 17 18
4 9 13 16 0 19 20
5 10 14 17 19 0 21
6 11 15 18 20 21 0

```

Матриця інцидентностей графа виражає кінцевий результат і виводиться на екран у вигляді таблиці кодованих об'єктів.

Кодований об'єкт представлено матрицею інцидентностей

```

1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 1 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0
0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 1 1 0 0 0 0
0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 1 0 0
0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1 0
0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 1 1 1

```

### 3.6 Інструкція користувачу

Даний програмний продукт реалізовано на мові високого рівня C++.  
Необхідні ресурси комп'ютера для нормальної роботи програмного забезпечення:

- а) тип процесора – Pentium II -200 МГ і вище;
- б) об'єм ОЗП – 32 МВ і більше;

- в) монітор – VGA,SVGA;
- г) відеоадаптер – S3 або аналогічний;
- д) вільне місце на жорсткому диску – не менше 10 MB.

Ініціалізація програми відбувається запуском файлу khvalko.exe. Для подальшої роботи програми потрібно виконати наступні кроки (перехід до наступного кроку здійснюється натисканням клавіші Enter).

1. Вводимо кількість вершин графа (кількість об'єктів). В нашому випадку вона рівна 7.
2. Вводимо кількість зв'язків між даними об'єктами. Вони розраховуються за формулою і в нашому випадку становлять 21.
3. Заповнюємо запропоновану матрицю суміжностей вагами зв'язків.
4. Після введення вхідних даних отримаємо матрицю інцидентностей (матрицю кодованих об'єктів).

Результати розрахунків, які були отримані в результаті машинного експерименту:

Матриця суміжності рангів графа:

```

0 1 2 3 4 5 6
1 0 7 8 9 10 11
2 7 0 12 13 14 15
3 8 12 0 16 17 18
4 9 13 16 0 19 20
5 10 14 17 19 0 21
6 11 15 18 20 21 0

```

Матриця інцидентностей графа виражає кінцевий результат і виводиться на екран у вигляді таблиці кодованих об'єктів.

Кодовані об'єкти представлена матрицею інцидентностей

```

1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 1 0 0 0 0 1 1 1 1 0 0 0 0 0 0
0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 1 1 1 0 0 0
0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 1 0
0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 1 0 1
0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 1 0 1 1 0

```

## ВИСНОВКИ

У процесі проведення досліджень у магістерській роботі вдалося досягти значних результатів, які сприятимуть розвитку області автоматизованого управління.

У першому розділі проведений аналіз методів опису інформації в системах автоматизованого управління. Виявлено, що ці методи переважно спрямовані на забезпечення завадостійкості в каналах зв'язку та синхронізацію обміну даними. Однак вони не сприяють підвищенню ефективності прийняття рішень з управління. На відміну від існуючих методів, у магістерській роботі вдалося довести переваги запропонованих рангових кодів. Ці коди спрямовані на уніфікацію алгоритмів опису та обробки інформації про стани об'єктів управління, що описані в різних параметричних просторах. Такий підхід дозволяє не лише покращити завадостійкість в обміні даними, але й підвищити загальну ефективність прийняття рішень в системах управління.

В другому розділі запропоновано і математично обгрунтовано методи опису даних ранговими конфігураціями і їх кодування ранговими кодами. Розроблено структуру пристрою передавання рангової інформації в комунікаційних мережах автоматизованих систем управління. Наведено обгрунтування побудови кодера інформації в рангову у вигляді нейронної мережі. Досліджено застосування рангового коду в нейронній мережі для розпізнавання цифр, що відкриває перспективи використання цього підходу в сучасних технологіях. Проведено експериментальні та теоретичні дослідження коригувальної здатності запропонованого рангового коду до локалізації і виправлення помилок, внесених в каналах зв'язку комунікаційних мереж, які показали можливість локалізації і виправлення симетричних помилок в кодових кодових словах і дозволили сформулювати вербальний алгоритм для її реалізації.

В третьому розділі були розроблені алгоритм кодування рангової інформації DRP-кодами, алгоритм декодування прийнятого DRP-коду в рангову конфігурацію, алгоритм локалізації і виправлення помилок в коді, внесених каналами комунікаційних мереж автоматизованих систем. Реалізація цих алгоритмів в програмному забезпеченні в середовищі програмування Turbo C показала їхню адекватність та ефективність у практичних застосуваннях. У результаті досліджень була висунута нова концепція рангових кодів, яка може знайти широке застосування в різних галузях автоматизованого управління та інформаційних технологій.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Быков Н.М., Пастушенко О.С., Коберский В.Г. Обобщенный метод принятия решений в системах управления и контроля. *Контроль и управление в технических системах*: 3 – я междунар. науч.-техн. конф. 18 - 21 сент. 1995 г.: тезисы докл. - I., 1995. - с. 223 - 224.
2. Биков М.М., Кузьмін І.В., Яковенко А.І. Кластеризація даних з використанням потенціальних кодів. *Вісник Вінницького політехнічного інституту*. - 2001. - № 6. - 61-64 с.
3. Биков М.М. Універсальний метод представлення інформації в інтелектуальних еволюційних системах. *Відбір і обробка інформації*. - 2006.- Вип. 24(100). С. 35-42.
4. Биков М.М., Конате К., Раїмі А. Теоретичні основи представлення і обробки інформації з використанням потенціальних кодів. *Вісник ВПІ*, - №2, - 2009 р., с. 88-98.
5. Биков М.М., Грищук Т.В., Кучерук Н.О. Метод підвищення пропускної здатності каналу за рахунок використання потенціальних кодів. В кн. *Методи та засоби кодування, захисту й ущільнення інформації. Тези доповідей другої Міжнародної конференції*. М. Вінниця, 22-24 квітня 2009 р. - Вінниця: Універсум-Вінниця, 2009. - с.41-42
6. Биков М.М., Філатова М.М. Визначення характеристик потенціальних кодів за моделями рангових конфігурацій. *Вісник Хмельницького національного університету*, №5, 2013 р.– с. 92-97.
7. Штейнберг С. Идентификация в системах управления. М.: Энергоатомиздат, 1987.–196 с.
8. Tou J., Gonzalez R. *Pattern Recognition Principle*. London-Amsterdam-Don Mill.: Addison-Wesley, 1974. – 411 p.
9. Kosko B. Fuzzy entropy and conditioning. *Information sciences* – 1966. – 2. - P.165-174.

10. Kruskal J. Multidimensional scaling by optimizing goodness-of-fit to a nonmetric hypothesis. *Psychometrika* – 1964. – 29. - P. 1-28, 115- 129.
11. Pawlak Z., Skowron A. Rough membership functions. *Advances in the Dempster-Shafer theory of evidence*; Ed. Yager R., Fedrizzi M., Kacprzyk J. - *New York: John Wiley & Sons*, 1994. – P. 251-271.
12. Дюбуа Д., Прад А. Теория возможностей. Приложения к представлению знаний в информатике: Радио и связь, 1990. – 288 с.
13. Ялковський А.Є. Проблеми розпізнавання мови людини. *Проблеми інформатизації та управління*. 2009. №27, С.163-166.
14. Карпов О.М. Методи аналізу і розпізнавання складних сигналів в автоматизованих системах мовного діалогу: дисертація д-ра техн. наук: 05.13.2006. Національний авіаційний ун-т. - К., 2003, – 216 С.
15. Патент України 71189 А, 7 G10L19/04. Спосіб розпізнавання образів. Биков М.М., Грищук Т.В., Філатова М.М. 71189 А. Бюлетень №11 від 15.11.2014.
16. Семенець Д.А. Моделювання людино-машинних систем управління з голосовим інтерфейсом: Дис. канд. техн. наук: 01.05.02. Вінницький національний технічний ун-т. — Вінниця, 2003. — 220 С.
17. Мартін Т. Практичне застосування мовного введення в обчислювальну машину. ТИИЭР. – 1976, т.64, N 6.
18. Gleve R.D., Giet von der G. The DAWID speech recognition system // R.D. Glave, G. Giet. - Proc.Int. Сокр. ASSP, Talsa,1978, p.429-432.
19. Preparata F.P., Nievergelt J. Difference-Preserving Codes // F.P. Preparata, J. Nievergelt. - IEE Trans. Information theory, 1974, v .20, p. 643-649.
20. Злотник Б.М. Помехоустойчивые коды в системах связи. Радио и связь, 1989.– 232 с.
21. Акулиничев Ю.П., Бернгардт А.С. Теория и техника передачи информации. Учебное пособие — Томск: Эль Контент, 2012. — 210 с.
22. Жураковський Ю.П., Полторак В.П. Теорія інформації і кодування: підручник. Вища школа, 2001. – 256 с.

- 23.Скляр Б. Цифровий зв'язок. Теоретичні основи і практичне застосування, 2-ге видд.: Пер. з англ. Вид. буд. "Вільямс", 2003. – 1104 с.
- 24.Камерон П., ван Линт Дж. Теория графов, теория кодирования и блок-схемы. Мир, 1980. - 144 с.
25. М. Биков, І. Кузьмін, О. Коберський. Порівняльна характеристика ефективності структурних кодів системах розпізнавання мови. В. кн. Труды V Міжнародної конференції «Контроль і управління в складних системах (КУСС-99)». Вінниця, ВНТУ, 1999. – С.191- 194.
26. Бойко Ю. М., Дружинінін В. А., Толюпа С. В. Теоретичні аспекти підвищення завадостійкості й ефективності обробки сигналів в радіотехнічних пристроях та засобах телекомунікаційних систем за наявності завад : монографія - Київ : Логос, 2018. - 227 с.
27. Конспект лекцій з навчально їдисципліни «Теорія інформації та кодування» / Укладач: С. В. Соколов. – Суми: Вид-во СумДУ, 2020. – 101 с.
28. Dit: discrete information theory — dit 1.2.3 documentation. Режим доступу : <http://docs.dit.io/en/latest/> (дата звернення: 2023-10-27).
29. Курко А.М. Введення в теорію інформації: Посібник до вивчення дисципліни «Теорія інформації». Тернопільський національний технічний університет ім. Івана Пулюя, 2017. – 108 с. – Режим доступу: <http://elartu.tntu.edu.ua/handle/lib/21919> (дата звернення: 2023-10-27).
30. Основи теорії інформації та кодування: лабораторний практикум: навч. посіб. для студ. спеціальності 171 «Електроніка», спеціалізації «Електронні та інформаційні технології кінематографії та аудіовізуальних систем» / М. І. Романюк, Г. Г. Власюк; КПІ ім. Ігоря Сікорського. – Київ : КПІ ім. Ігоря Сікорського, 2018. – 81 с. (дата звернення: 2023-10-30).
31. Coding theory: the first 50 years | plus.maths.org. Режим доступу : <https://plus.maths.org/content/coding-theory-first-50-years> (дата звернення: 2023-10-27).

## **ДОДАТКИ**



Додаток А  
(обов'язковий)

Міністерство освіти і науки України  
Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ

\_\_\_\_\_ д.т.н., проф. Олег

БІСІКАЛО

«\_\_» \_\_\_\_\_ 2023 року

ТЕХНІЧНЕ ЗАВДАННЯ  
на магістерську кваліфікаційну роботу  
РОЗРОБКА І ДОСЛІДЖЕННЯ ІНТЕЛЕКТУАЛЬНИХ МЕТОДІВ ОПИСУ  
ІНФОРМАЦІЇ В КОМУНІКАЦІЙНИХ МЕРЕЖАХ АВТОМАТИЗОВАНИХ  
СИСТЕМ

08-31.МКР.007.02.000 ТЗ

Керівник: к.т.н., професор каф. КСУ

\_\_\_\_\_ Микола БИКОВ

«\_\_» \_\_\_\_\_ 2023 р.

Розробив студент гр. ЗАКІТ-22м

\_\_\_\_\_ Андрій ЗАПЛЕТНЮК

«\_\_» \_\_\_\_\_ 2023 р.

Вінниця 2023

### 1. Назва та галузь застосування

Розробка і дослідження інтелектуальних методів опису інформації в комунікаційних мережах автоматизованих систем. Системи прийняття рішень, інформаційні системи і Інтернет речей.

### 2. Підстава для проведення розробки.

Підставою для виконання роботи є наказ №\_\_ по ВНТУ від «\_\_» \_\_\_\_\_ 2023р., та індивідуальне завдання на МКР, затверджене протоколом №\_\_ засідання кафедри АІТ від «\_\_» \_\_\_\_\_ 2023р.

### 3. Мета та призначення розробки.

Підвищення ефективності прийняття рішень в системах управління за рахунок розробки методів, моделей і пристроїв представлення і передачі інформації в системах управління, ефективних щодо обчислювальних затрат та способів корекції помилок.

### 4. Джерела розробки.

Магістерська дипломна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Основы теории информации та кодирования / І.В.Кузьмін І.В. Троцишин, А.І. Кузьмін, В.О. Кедрус, В.Р. Любчик. 3-тє вид., перероб. та допов. - Хмельницький: ХНУ, 2009. - 373 с.
2. Биков М.М. Універсальний метод представлення інформації в інтелектуальних еволюційних системах / Биков М.М. // Відбір і обробка інформації. - 2006.- Вип. 24(100). С. 35-42.
3. Биков М.М. Кластеризація даних з використанням потенціальних кодів / М.М. Биков, І.В. Кузьмін, А.І. Яковенко // Вісник Вінницького політехнічного інституту. - 2001. - № 6. - 61-64 с.
4. Биков М.М., Конате К., Раїмі А. Теоретичні основи представлення і обробки інформації з використанням потенціальних кодів. // Вісник ВПІ, - №2, - 2009 р., с. 88-98.

### 5. Вимоги до розробки.

#### 5.1. Перелік головних функцій:

- обчислення кількості рангових конфігурацій для дослідження інтервальної моделі;
- кодування рангової інформації в ранговий код;
- декодування рангового коду в рангову інформацію.

#### 5.2. Основні технічні вимоги до розробки.

##### 5.2.1. Вимоги до програмної платформи:

- WINDOWS 7\10;
- Microsoft Visual Studio 17;

##### 5.2.2. Умови експлуатації системи:

- робота на стандартних ПК в приміщеннях зі стандартними умовами;
- можливість цілодобового функціонування системи;
- текст програмного забезпечення системи є цілком закритим.

## 6. Стадії та етапи розробки.

### 6.1 Пояснювальна записка:

1	Техніко-економічне обґрунтування доцільності розробки	02.10.2023 р.
2	Постановка задачі і розробка технічного завдання	04.10. 2023 р.
3	Математичне обґрунтування вибраного методу представлення інформації	20.10. 2023 р.
4	Розробка алгоритмічного та програмного забезпечення системи	15.11. 2023 р.
5	Машинний експеримент	27.11. 2023 р.
6	Оформлення пояснювальної записки, графічного матеріалу і презентації	30.11. 2023 р.
7	Апробація результатів дослідження	07.12.2023 р.
8	Публікації	10.12.2023 р.
7.2	Графічні матеріали:	
	Плакати демонстраційні	01.12. 2023 р.
	Схеми програм дослідження інтервальної моделі	01.12. 2023 р.
	Схема пристрою передачі інформації	02.12. 2023 р.
	Схема декодера	02.12. 2023 р.
	Схема програми кодування рангової інформації	04.12. 2023 р.
	Схема програми декодування рангової інформації	05.12. 2023 р.
	Результати машиноого експерименту	06.12. 2023 р.
10	Захист МКР	12.12. 2023 р.

## 7. Порядок контролю і приймання.

- 7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до « 01 » листопада 2023 р.
- 7.2. Атестація проекту здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до « 21 » листопада 2023 р.
- 7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ДЕК. Захист магістерської дипломної роботи провести до « 12 » грудня 2023 р.

Додаток Б  
(обов'язковий)

## **ІЛЮСТРАТИВНА ЧАСТИНА**

РОЗРОБКА І ДОСЛІДЖЕННЯ ІНТЕЛЕКТУАЛЬНИХ МЕТОДІВ ОПИСУ  
ІНФОРМАЦІЇ В КОМУНІКАЦІЙНИХ МЕРЕЖАХ АВТОМАТИЗОВАНИХ  
СИСТЕМ

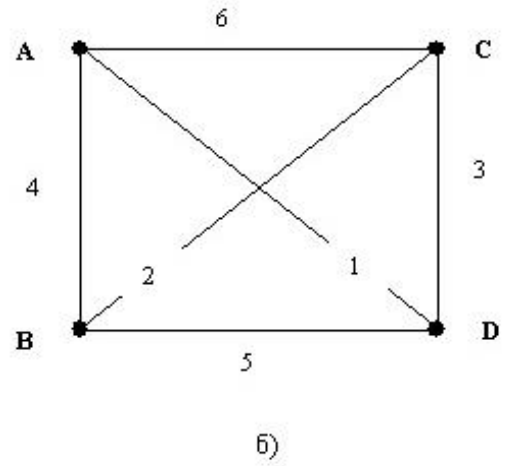
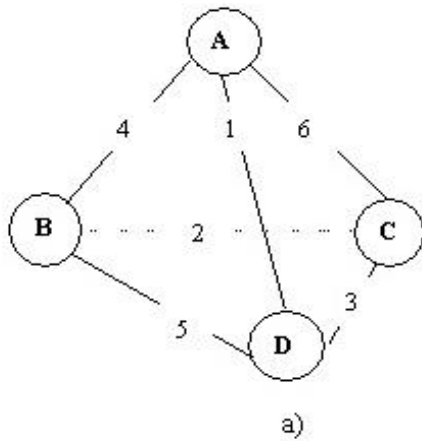


Рисунок Б.1 - Геометричні моделі рангової конфігурації:  
 а – тривимірний симплекс; б – повний регулярний граф.

**Алгебраїчна модель рангової конфігурації:**

$$K = \{ \{1,4,6\}, \{2,4,5\}, \{2,3,6\}, 1,3,5 \}$$

Ранги

	6	5	4	3	2	1
А	1	0	1	0	0	1
В	0	1	1	0	1	0
С	1	0	0	1	1	0
Д	0	1	0	1	0	1

	А	В	С	Д
А		4	6	1
В	4		2	5
С	6	2		3
Д	1	5	3	

О  
Б'  
е  
к  
т  
и

а)

б)

Рисунок Б.2 - Комбінаторні моделі рангової конфігурації:

а) матриця інцидентностей графа

б) матриця суміжностей рангів для графа

**ВИЗНАЧЕННЯ РАНГОВОГО КОДУ**

Код  $v$ , який зберігає ранги відстаней ( $DRP$ -код), є відображення  $i \rightarrow v_i$  множини  $M = \{1, 2, \dots, m\}$  в множину  $\{0,1\}^n$  двійкових послідовностей довжини  $n$  таке, що

$$\forall_{i,j} (R(d_{ij}) = r \Rightarrow R(h_{ij}) = r), \quad r = \overline{1, m_r}, \quad i, j \in M .$$

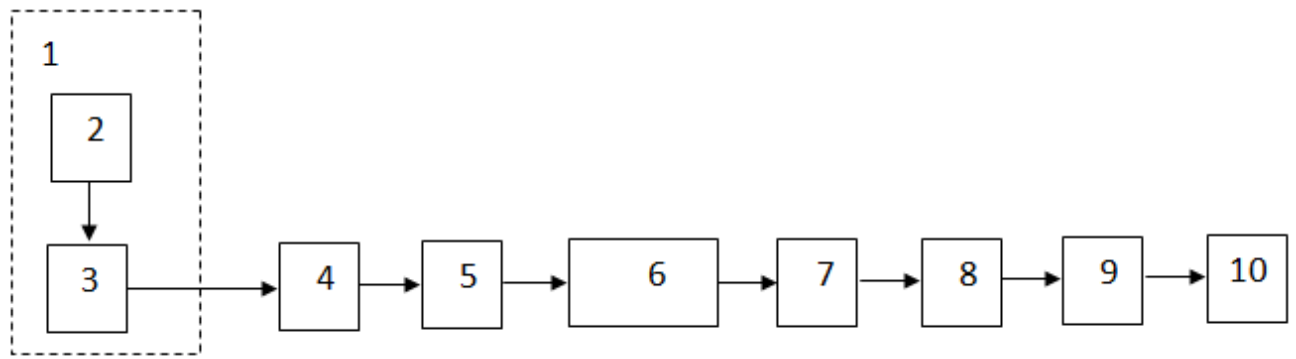


Рисунок Б.3 – Структурна схема пристрою передачі інформації

1 - джерело інформації,

2 -джерело повідомлень (група елементів або об'єктів);

3- кодер джерела інформації;

4 - кодер каналу;

5 - модулятор сигналу;

6 -канал передачі інформації;

7 - приймача сигналу;

8 – демодулятор сигналу;

9 - декодер DRP-коду;

10 - блоком прийняття рішень.

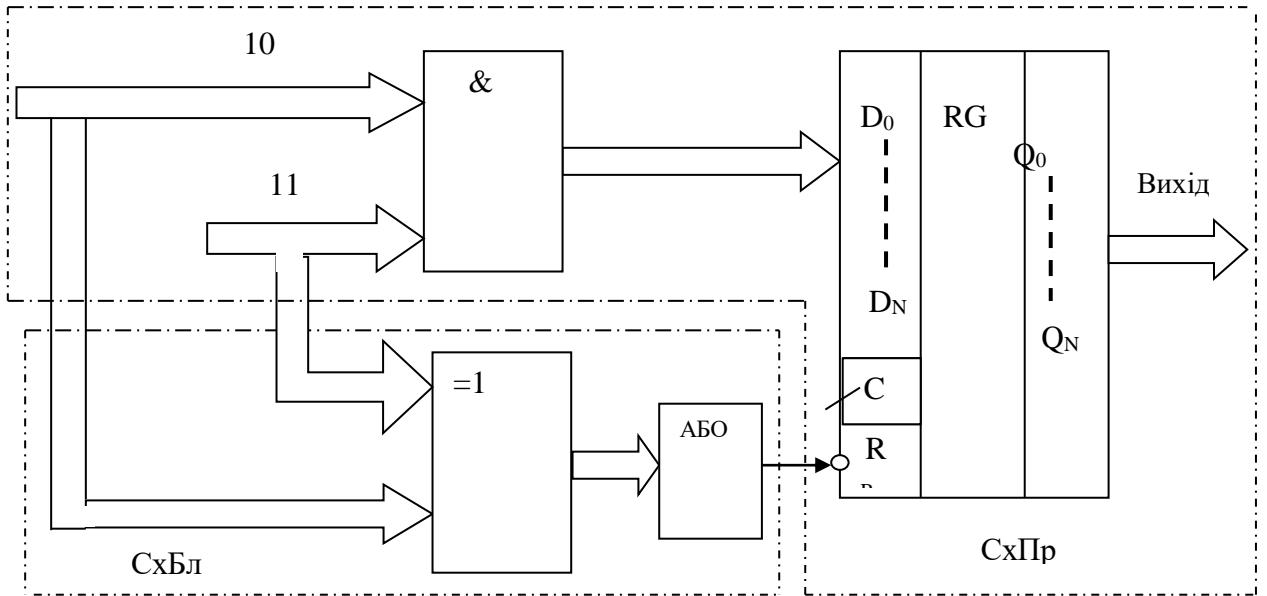


Рисунок Б.4 – Функціональна схема декодера

## РАНГОВА МОДЕЛЬ НЕЙРОННОЇ МЕРЕЖІ

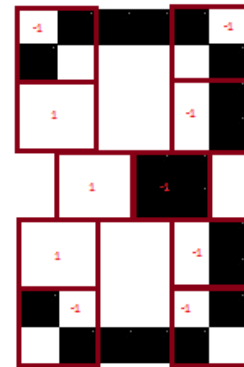
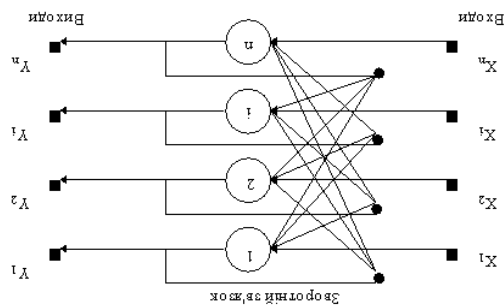


Рисунок Б.5 - Архітектура нейронної мережі Хопфілда

Рисунок Б.6 - Схема розташування площадок-ознак на рисунку цифри 3

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	26	0	0	0	0	0	0	0	0	0	0	0
3	44	36	0	0	0	0	0	0	0	0	0	0
4	3	35	36	0	0	0	0	0	0	0	0	0
5	33	16	7	17	0	0	0	0	0	0	0	0
6	41	11	19	28	31	0	0	0	0	0	0	0
7	9	32	43	30	18	34	0	0	0	0	0	0
8	41	19	9	10	11	44	9	0	0	0	0	0
9	37	43	13	26	26	20	4	42	0	0	0	0
10	30	30	3	44	6	14	17	36	22	0	0	0
11	30	5	7	40	13	39	45	15	35	21	0	0
12	25	36	4	44	4	18	13	28	22	12	36	0
13	7	32	33	29	18	34	11	7	15	9	35	0
14	28	5	4	19	40	29	7	1	18	32	45	0
15	17	10	24	20	28	39	41	19	32	6	43	0
16	20	43	36	1	17	18	20	41	41	7	33	0
17	27	10	4	16	30	45	26	8	34	32	44	0
18	40	2	30	35	28	15	1	14	7	19	6	0

Рисунок Б.7 - Приклад матриці рангів ваг міжнейронних зв'язків

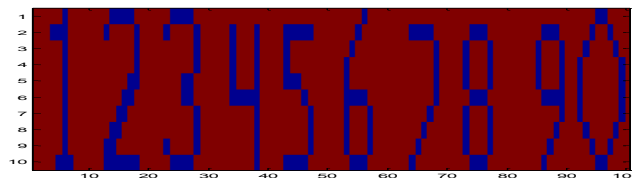


Рисунок Б.8 - Еталонні зображення цифр



Ранги → Символи ↓	6	5	4	3	2	1
A	1	0	1	0	0	1
B	0	1	1	0	1	0
C	1	0	0	1	1	0
D	0	1	0	1	0	1

а)

Ранги → Відстані ↓	6	5	4	3	2	1
d(AC)	1	0	0	0	0	0
d(AB)	0	0	1	0	0	0
d(AD)	0	0	0	0	0	1
d(BC)	0	0	0	0	1	0
d(BD)	0	1	0	0	0	0
d(CD)	0	0	0	1	0	0

б)

Рисунок Б.9 - Правильний DRP-код рангової конфігурації (а) і перевірна матриця рангових відстаней для нього (б)

Ранги → Символи ↓	6	5	4	3	2	1
A	1	0	1	0	0	1
B	1	0	1	0	1	1
C	1	0	0	1	1	0
D	0	1	0	1	0	1

а)

Ранги → Відстані ↓	6	5	4	3	2	1
d(AC)	1	0	0	0	0	0
d(AB)	1	0	1	0	0	0
d(AD)	0	0	0	0	0	1
d(BC)	1	0	0	0	1	0
d(BD)	0	0	0	0	0	0
d(CD)	0	0	0	1	0	0

б)

Рисунок Б.10 - DRP-код рангової конфігурації (а) з помилкою в кодовому слові В ("1" з розряду 5 перейшла в 6 розряд, а на її місці став "0") і перевірна матриця рангових відстаней для нього (б)

## ТЕОРЕТИЧНА ОЦІНКА КОРИГУВАЛЬНОЇ ЗДАТНОСТІ РАНГОВИХ КОДІВ

Теорема: для рангової конфігурації розмірності  $m$  відстань Хемінга  $d_x$  між словами рангового коду дорівнює:

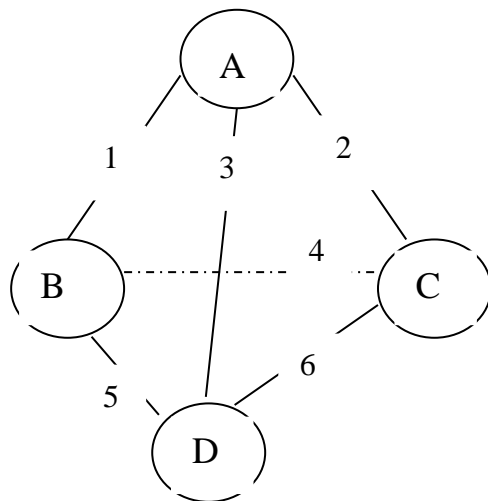
$$d_x = 2(m-1) - 2.$$

теоретична оцінка можливості виявлення помилок кратністю  $g_d$

$$g_d \leq d_x - 1 = 2(m-1) - 3$$

теоретична оцінка коригування помилок кратністю  $g_c$  – за формулою:

$$g_c \leq \frac{(d_x - 1)}{2} = \frac{(2(m-1) - 3)}{2}$$



Кодова- ні сим- воли	Ранги					
	6	5	4	3	2	1
A:	0	0	0	1	1	1
B:	0	1	1	0	0	1
C:	1	0	1	0	1	0
D:	1	1	0	1	0	0

Рисунок Б.11 - Модель рангової конфігурації для  $m=4$ : а) – ранговий симплекс; б) – матриця інцидентності

$$d_x = 2(4-1) - 2 = 4$$

$$g_d \leq d_x - 1 = 4 - 1 = 3$$

$$g_c \leq \frac{(d_x - 1)}{2} = \frac{3}{2} = 1$$

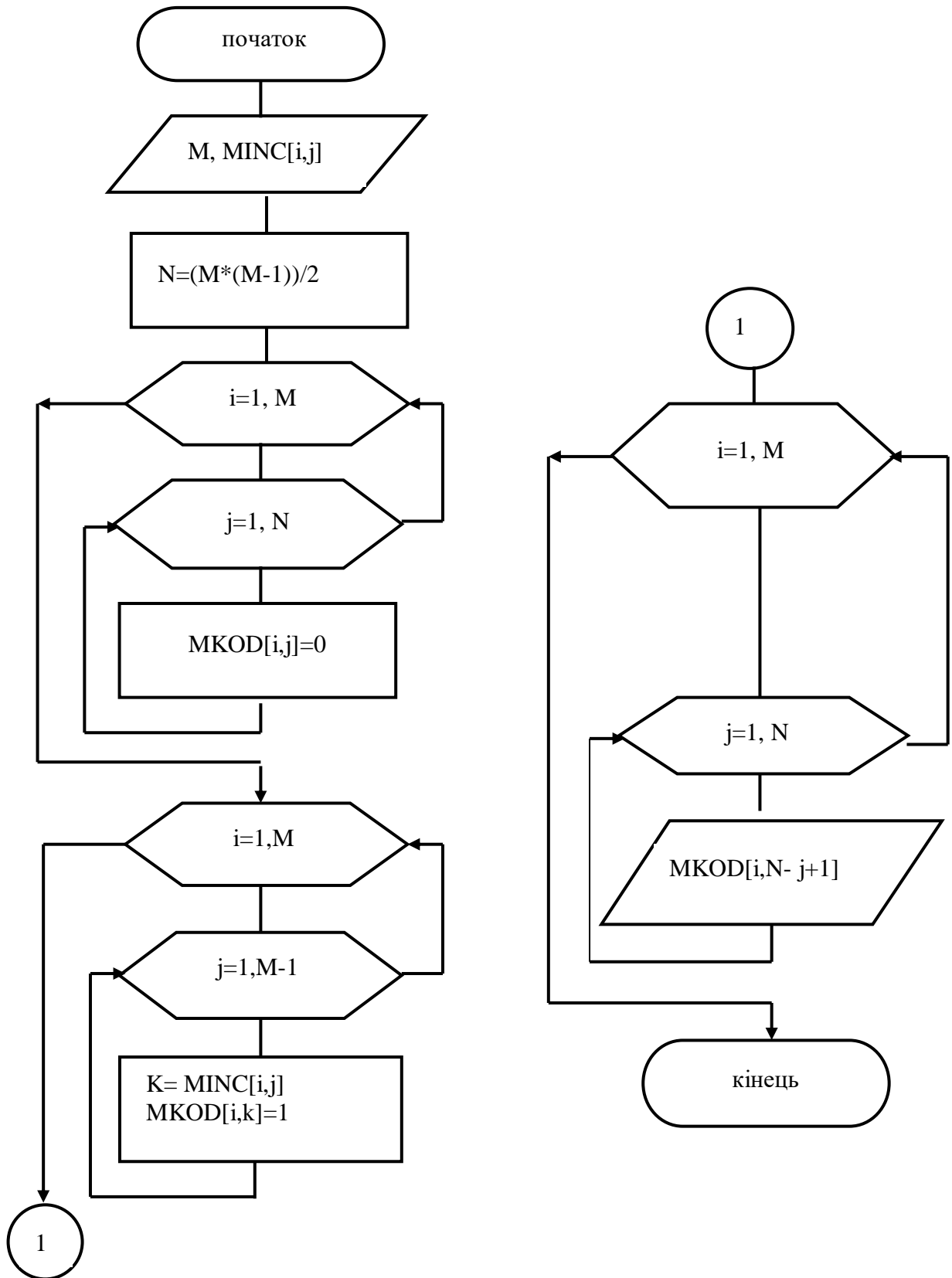


Рисунок Б.12 – Алгоритм кодування рангової конфігурації DRP-кодом

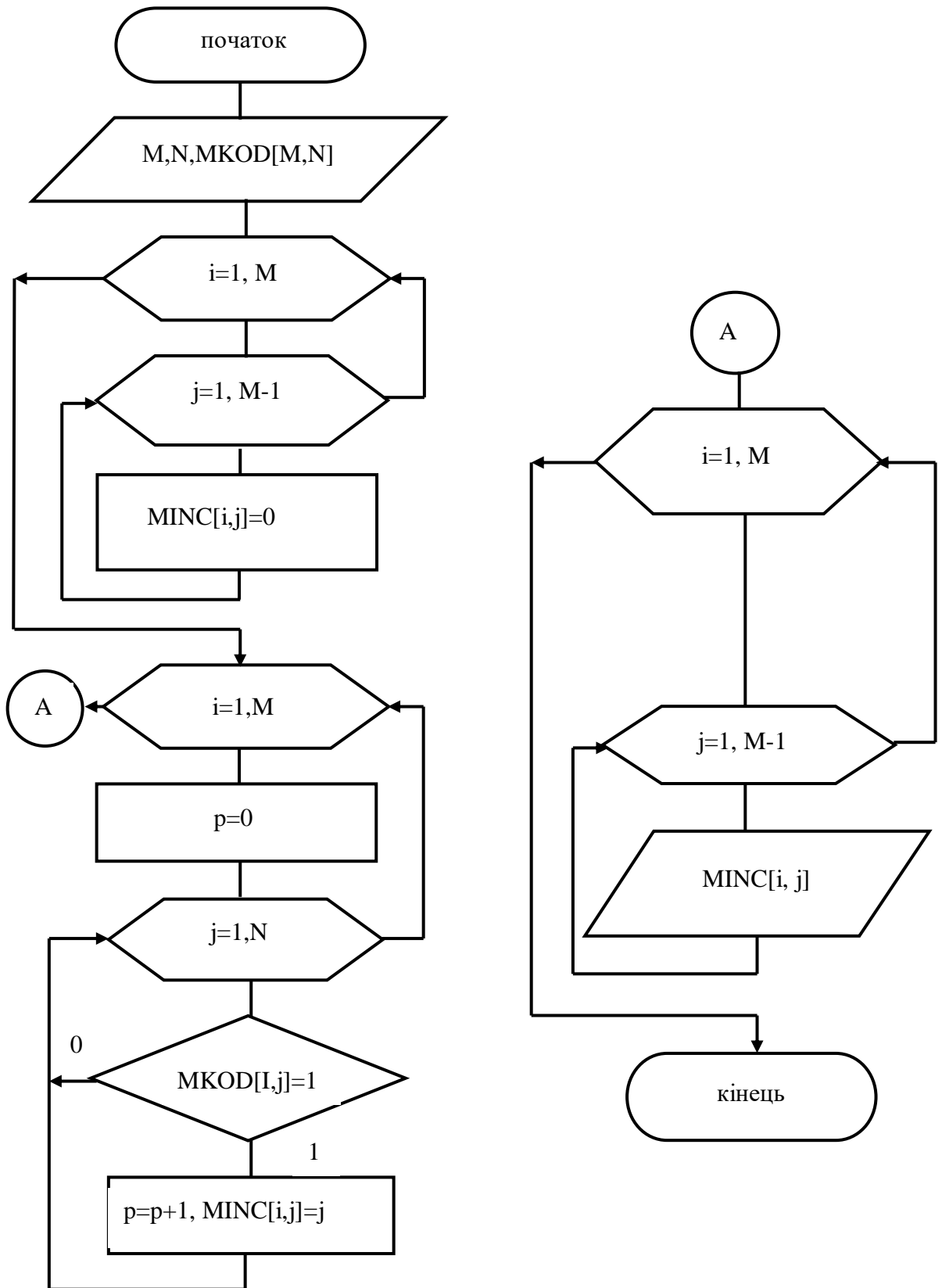


Рисунок Б.13 – Схема алгоритму роботи декодера

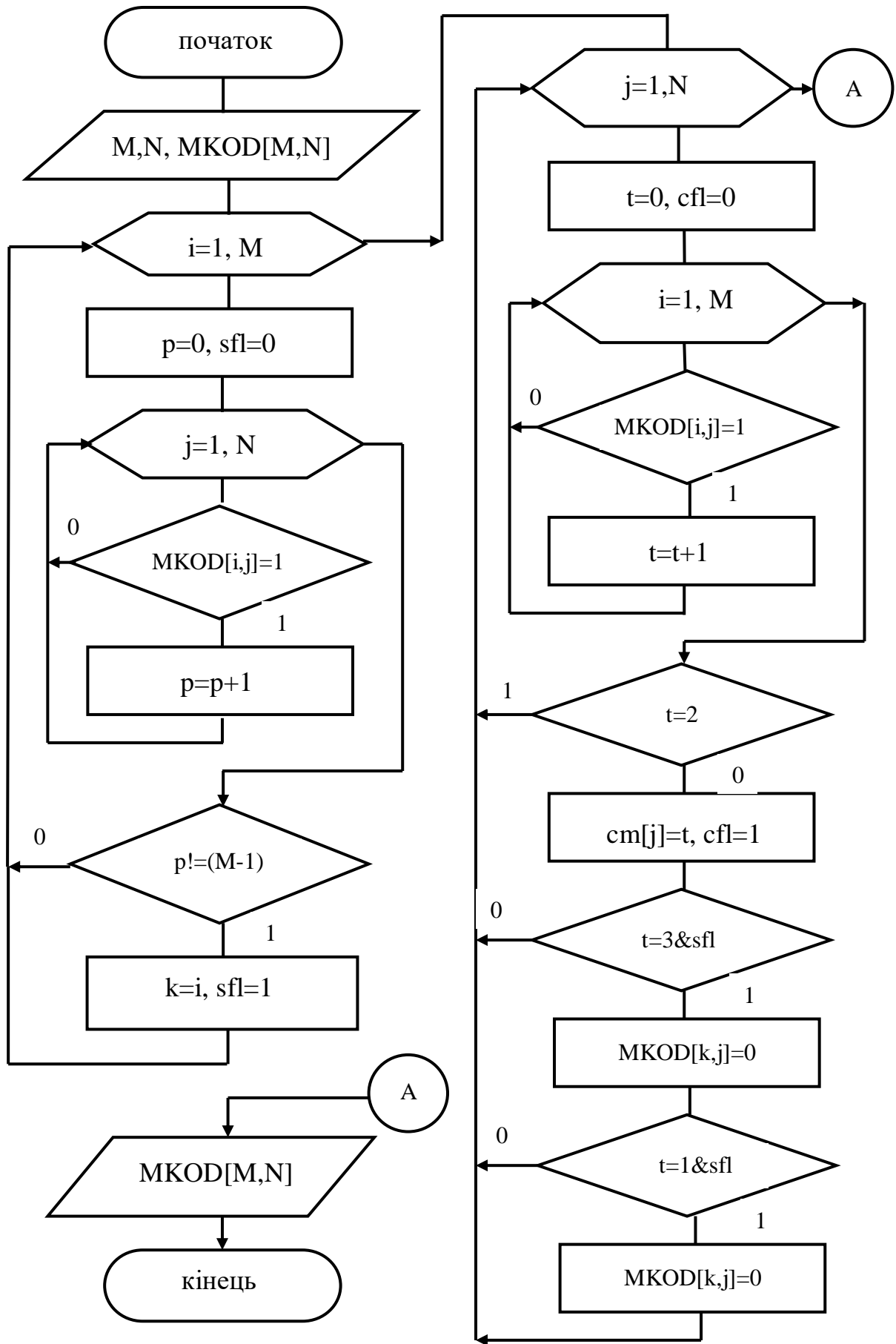
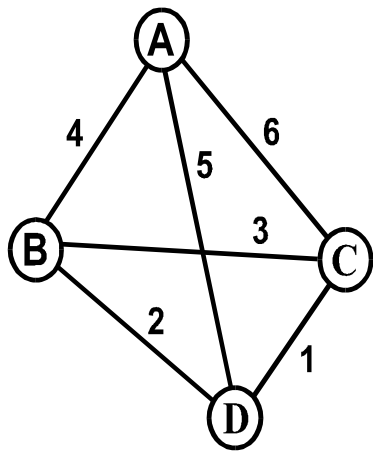
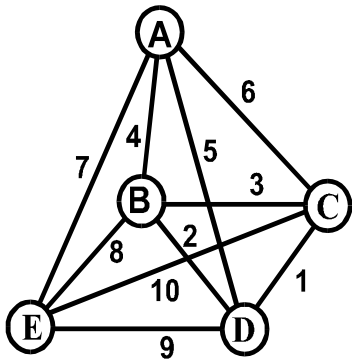


Рисунок Б.14 – Алгоритм виявлення та корегування помилок



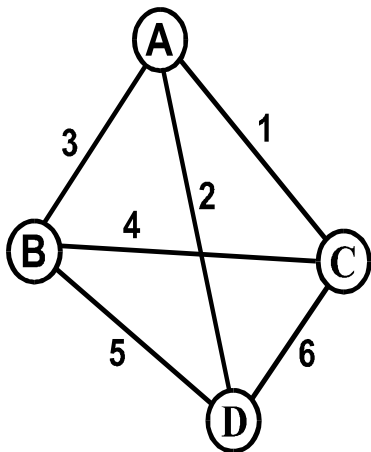
4	5	6
2	3	4
1	3	6
1	2	5

O \ R	6	5	4	3	2	1
A	1	1	1	0	0	0
B	0	0	1	1	1	0
C	1	0	0	1	0	1
D	0	1	0	0	1	1



4	6	5	7
4	3	2	8
6	3	1	10
5	10	1	9
7	8	10	9

O \ R	10	9	8	7	6	5	4	3	2	1
A	0	0	0	1	1	1	1	0	0	0
B	0	0	1	0	0	0	1	1	1	0
C	1	0	0	0	1	0	0	1	0	1
D	1	1	0	0	0	1	0	0	0	1
E	1	1	1	1	0	0	0	0	0	0



3	1	2
3	4	5
1	4	6
2	5	6

O \ R	6	5	4	3	2	1
A	0	0	0	1	1	1
B	0	1	1	1	0	0
C	1	0	1	0	0	1
D	1	1	0	0	1	0

Рисунок Б.15 – Приклад рангових конфігурацій та рангових кодів

Додаток В  
(обов'язковий)  
Лістинги програм

В 2.1 Програма кодування рангової конфігурації

```
#include<stdlib.h>
#include<conio.h>
#include <stdio.h>

int main()
{
    int nSize, nSizeVaga;
    int i,j;
        clrscr();

printf("Vvedit kilkist vershun:");
        scanf("%d",&nSize);

printf("Vvedit kilkist vag:");
        scanf("%d",&nSizeVaga);

int **mas = (int**)malloc(1000);
        for(i=0;i<nSize;i++)
            mas[i]=(int*)malloc(1000);

int **masVuh = (int**)malloc(1000);
        for(i=0;i<nSizeVaga;i++)
            masVuh[i]=(int*)malloc(1000);

for(i=0;i<nSize;i++)
    {
        for(j=0;j<nSizeVaga;j++)
            masVuh[i][j] = 0;
    }
printf("\nZapovnennya matruci:");

for(i=0;i<nSize;i++)
    {
        for(j=0;j<nSize;j++)
            {
                scanf("%d",&mas[i][j]);
```

```

        }
    }
    printf("\nVhidna matrucya:");

    for(i=0;i<nSize;i++)
    {
        printf("\n");
        for(j=0;j<nSize;j++)
        {
            printf(" %1.1d ",mas[i][j]);
        }
    }
    for(i=0;i<nSize;i++)
    {
        for(j=0;j<nSize;j++)
        {
            if(mas[i][j]>0)
            {
                masVuh[i][mas[i][j]-1] = 1;
            }
        }
    }
    printf("\nVuhidna matrucya:");

    for(i=0;i<nSize;i++)
    {
        printf("\n");
        for(j=0;j<nSizeVaga;j++)
        {
            printf(" %d ",masVuh[i][j]);
        }
    }

    getch();
    return 0;
}

```

## В 2.2 Програма декодування DRP-коду

```

#include<conio.h>
#include<stdlib.h>
#include<stdio.h>
#include<math.h>

```



```

#define m 4
int i,j,t,r,x,c,k,v,b; float w;    float kol; float pod;
int temp[m*(m-1)/2+1];
int temp1[m*(m-1)/2+1];
int temp2[802][m+1][m+1];
FILE *f;
void zapovnena(int temp[m*(m-1)/2+1],int a[m+1][m+1]);
void sumetria(int a[m+1][m+1]);
void peretvor_matrits(int a[m+1][m+1],int a1[m+1][m+1]);
int porivniana(int k,int temp2[802][m+1][m+1], int a1[m+1][m+1]);
void sort(int a1[m+1][m+1]);
void writel(int a[m+1][m+1]);
void readl(int n,int s,int temp2[802][m+1][m+1]);
void main()
{
f=fopen("mas.dat","w");
fclose(f);
int a[m+1][m+1]; int a1[m+1][m+1]; int a2[m+1][m+1]; int s;
clrscr();
w=0;
for(j=0;j<m*(m-1)/2+1;j++)
temp[j]=j;
i=1; kol=0;pod=0;
while(i!=0)
{
kol++;
zapovnena(temp,a);
sumetria(a);
peretvor_matrits(a,a1);
if(kol==1)
{
for(v=1;v<m+1;v++)
for(b=1;b<m;b++)
a2[v][b]=a1[v][b];
sort(a2);
writel(a2);
}
}
}

```

```

    }
    sort(a1);
    read1(w,0,temp2);
    for(k=0;k<w;k++)
    {
        pod+=porivniana(k,temp2,a1);
        if(porivniana(k,temp2,a1)==1)
            break;
    }
    if(porivniana(k,temp2,a1)==0)
        writel(a1);
    i=m*(m-1)/2-1;
    while(temp[i]>temp[i+1])
        i=i-1;
    j=m*(m-1)/2;
    while(temp[i]>temp[j])
        j=j-1;
    c=temp[i];
    temp[i]=temp[j];
    temp[j]=c;
    r=m*(m-1)/2;
    x=i+1;
    while(r>x)
    {
        c=temp[r];
        temp[r]=temp[x];
        temp[x]=c;
        r=r-1;
        x=x+1;
    }
}

printf ("\n кількість перестановок=\t\t%10.0f", kol);
printf ("\n\n кількість подібних матриць=\t%10.0f", pod-1);
printf ("\n\n кількість різних матриць=\t%10.0f", kol-pod+1);
getch();
}

```

```
// підпрограма-функція формування матриці з одномірного масиву
void zapovnena(int temp[m*(m-1)/2+1],int a[m+1][m+1])
{
k=1;
for(v=1;v<m;v++)
    for(b=v+1;b<m+1;b++)
        {
            a[v][b]=temp[k];
            k++;
        }
}
// підпрограма-функція формування кососиметричної матриці
void sumetria(int a[m+1][m+1])
{
    for(v=1;v<m;v++)
        for(b=2;b<m+1;b++)
            {
                a[b][v]=a[v][b];
                a[v][v]=0;
                a[b][b]=0;
            }
}
// підпрограма-функція формування звичайно матриці кососиметричної

void peretvor_matrits(int a[m+1][m+1],int a1[m+1][m+1])
{
for(v=1;v<m+1;v++)
    for(b=1,t=1;b<m+1,t<m+1;b++,t++)
        {
            if(v==b)
                {
                    t--;
                    continue;
                }
            a1[v][t]=a[v][b];
        }
}
```

```
}  
int porivniania(int k,int temp2[802][m+1][m+1], int a1[m+1][m+1])  
{  
    for(v=1;v<m+1;v++)  
        for(b=1;b<m;b++)  
            if(temp2[k][v][b]!=a1[v][b])  
                return 0;  
    return 1;  
}  
void sort(int a1[m+1][m+1])  
{  
    for(v=1;v<m+1;v++)  
        for(k=1;k<m-1;k++)  
            {  
                c=a1[v][k];  
                for(b=k+1;b<m;b++)  
                    if(a1[v][b]<c)  
                        {  
                            c=a1[v][b];  
                            a1[v][b]=a1[v][k];  
                            a1[v][k]=c;  
                        }  
            }  
    for(v=1;v<m;v++)  
        {  
            for(b=v+1;b<m+1;b++)  
                {  
                    c=a1[v][1];  
                    if(a1[b][1]<c)  
                        {  
                            for(t=1;t<m;t++)  
                                {  
                                    temp1[t]=a1[b][t];  
                                    a1[b][t]=a1[v][t];  
                                    a1[v][t]=temp1[t];  
                                }  
                        }  
                }  
        }  
}
```

```

        continue;
    }
    if(a1[b][1]>c)
        continue;
    for(k=2;k<m;k++)
    {
        if(a1[b][k]<a1[v][k])
        {
            for(t=1;t<m;t++)
            {
                temp1[t]=a1[b][t];
                a1[b][t]=a1[v][t];
                a1[v][t]=temp1[t];
            }
            break;
        }
        if (a1[b][k]==a1[v][k])
            continue;
        break;
    }
}
}
void writel(int a[m+1][m+1])
{
    f=fopen("mas.dat","a");
    for(v=1;v<m+1;v++)
    {
        for(b=1;b<m;b++)
            fprintf(f," %d ",a[v][b]);
        fprintf(f,"\n");
    }
    w++;
    fclose(f);
}
void readl(int n,int s,int temp2[802][m+1][m+1])

```

```
{
    f=fopen("mas.dat","r");
        for(v=0;v<s;v++)
            {
                for(b=1;b<m;b++)
                    fscanf(f," %d",&t);
                fscanf(f,"\n");
            }
        for(c=0;c<n;c++)
            for(v=1;v<m+1;v++)
                {
                    for(b=1;b<m;b++)
                        fscanf(f," %d",&temp2[c][v][b]);
                    fscanf(f,"\n");
                }
    fclose(f);
}
```

Додаток Г  
(обов'язковий)

## ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: «Розробка і дослідження інтелектуальних методів опису інформації в комунікаційних мережах автоматизованих систем»

Тип роботи: Магістерська дипломна робота  
(БДР, МКР)

Підрозділ АІТ, ФІТА  
(кафедра, факультет)

### Показники звіту подібності Unicheck

Оригінальність \_\_\_\_\_ Схожість \_\_\_\_\_

Аналіз звіту подібності (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку \_\_\_\_\_ Роман МАСЛІЙ  
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи \_\_\_\_\_ Андрій ЗАПЛЕТНЮК  
(підпис) (прізвище, ініціали)

Керівник роботи \_\_\_\_\_ Микола БИКОВ  
(підпис) (прізвище, ініціали)