

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ЧЕРКАСЬКИЙ
ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Кваліфікаційна наукова праця
на правах рукопису

Скуцький Артем Борисович

ДИСЕРТАЦІЯ

МЕТОД І МОДЕЛІ СИСТЕМИ ЗАХИЩЕНОГО ІНФОРМАЦІЙНОГО
ОБМІНУ З НЕРОЗДІЛЬНИМ ФАКТОРІАЛЬНИМ КОДУВАННЯМ ДАНИХ

123 – Комп'ютерна інженерія

Подається на здобуття ступеня доктора філософії

Дисертація містить результати
власних досліджень. Використання
ідей, результатів і текстів інших
авторів мають посилання на
відповідне джерело

А.Б. СКУЦЬКИЙ

Науковий керівник:

Фауре Еміль Віталійович

доктор технічних наук, професор

Черкаси – 2025

АНОТАЦІЯ

Скуцький А.Б. Метод і моделі системи захищеного інформаційного обміну з нероздільним факторіальним кодуванням даних. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 123 – Комп'ютерна інженерія. – Черкаський державний технологічний університет, Черкаси, 2025.

Дисертаційна робота присвячена побудові системи захищеного інформаційного обміну з нероздільним факторіальним кодуванням даних (НФКД) в умовах зашумлених каналів зв'язку. Зростання обсягів передавання конфіденційних даних у відкритих каналах зв'язку потребує удосконалення методів синхронізації та шифрування з метою запобігання несанкціонованому доступу, забезпечення цілісності й достовірності інформації. Проблема синхронізації набуває особливої важливості в умовах, коли сторонній вплив або атаки можуть призвести до збоїв у встановленні синхронізму, що порушує працездатність системи. Відповідно, постає задача забезпечення організації захищеного достовірного передавання інформації за невідомого початкового моменту передавання даних каналами зв'язку з високим рівнем шуму. Для вирішення цієї задачі проведено аналіз існуючих методів кадрової синхронізації з використанням НФКД, що дозволив визначити подальший напрямок дослідження і сформулювати задачі роботи.

Розглянута математична модель процесу виявлення синхрокомбінацій у системах передавання даних з НФКД. Розглянуто два методи кадрової синхронізації: на основі поділу кодового слова на префіксну і суфіксну частини, а також на основі кореляційної обробки. Обидва методи використовують мажоритарну обробку бітів для підвищення достовірності приймання. Метод кадрової синхронізації на основі кореляційної обробки та метод достовірного передавання перестановок ґрунтується на виборі таких перестановок, що мають максимальне мінімальне значення відстані Хеммінга між їх двійковими представленнями та їх циклічними зсувами, що забезпечує

підвищення стійкості синхронізації за умов високої інтенсивності шуму. Розглянуто математичні моделі, що оцінюють ймовірність правильної та хибної синхронізації залежно від методу кадрової синхронізації. Наведено опис методу надійного передавання перестановок у зашумлених каналах з метою його використання в інформаційному обміні з НФКД та процедурою кадрової синхронізації.

Визначено обмеження існуючих методів кадрової синхронізації для нероздільних факторіальних кодів. Зокрема, підкреслено питання невизначеного моменту початку передавання синхрокомбінацій у каналах зв'язку з високою ймовірністю бітових помилок, що може призводити до хибних спрацювань або аварійного режиму приймача за фактичної відсутності передавання. Реакція приймача у таких системах має невизначеність через вплив шуму, що підвищує ймовірність некоректного встановлення синхронізму.

Аналіз робіт, присвячених матричним криптографічним перетворенням, створив передумови для використання матричних структур сумісно з НФКД. Зокрема, визначено актуальну задачу обґрунтування підходів до побудови скінченних полів квадратних матриць порядку 2 та подальшого їх застосування для узгодження ключів-перестановок у системах з НФКД. Розглянуто сучасні напрями розвитку матричних протоколів шифрування та узгодження ключів, заснованих на використанні математичних структур у криптографії. Описано переваги матричних методів, зокрема високу обчислювальну складність задачі зворотного перетворення, що забезпечує стійкість до дешифрування без ключа. Проаналізовано застосування скінченних полів, як теоретичної бази криптографічних алгоритмів, включаючи їхню роль у симетричному шифруванні, перевірці простоти чисел, факторизації та формуванні електронного цифрового підпису. Підкреслено перспективність використання скінченних полів матриць у криптографічних схемах, водночас вказано на відсутність бієктивного відображення матриць у перестановки, що створює виклик для інтеграції матричних і протоколів з

НФКД. Зазначено потребу в формуванні та дослідженні алгоритмів такого відображення. Підкреслено високий потенціал матричних підходів у криптографії та окреслено напрями подальших теоретичних і прикладних досліджень.

Об'єктом дослідження є процес захищеного передавання перестановок в умовах високої інтенсивності шуму в каналі зв'язку. Предметом дослідження визначено метод і моделі системи захищеного інформаційного обміну з НФКД.

Метою роботи є забезпечення захищеного інформаційного обміну в системах з НФКД зашумленим каналом зв'язку. Підвищення ефективності кадрової синхронізації системи інформаційного обміну в умовах високого рівня шуму досягнуто шляхом подальшого розвитку математичної моделі і методу виявлення синхрокомбінації за невідомого початкового моменту приймання даних передавача в каналі з високою ймовірністю бітових помилок, а також створення імітаційних моделей системи інформаційного обміну на основі НФКД з метою експериментального оцінювання отриманих теоретичних результатів.

Розроблено алгоритми перетворення текстових повідомлень у перестановки, що були використані при побудові макетних зразків інформаційного обміну текстовими повідомленнями через відкритий канал зв'язку industrial, scientific and medical band (ISM) діапазону. Наведені структури макетних зразків системи захищеного обміну текстовими повідомленнями на основі НФКД та їх апаратні складові. Отримані макетні зразки системи можуть бути використані, як навчальний стенд чи основа для Інтернету речей (IoT).

Наукова новизна дисертаційної роботи полягає в подальшому розвитку методу кадрової синхронізації, що використовує мажоритарну та кореляційну обробку прийнятих фрагментів синхрокомбінації, представленої у вигляді перестановки. Передумовою розвитку методу кадрової синхронізації став подальший розвиток математичної моделі процесу виявлення

синхрокомбінацій, що дозволяє оцінити ймовірність правильної та хибної синхронізації з застосуванням мажоритарної і кореляційної обробки за використання ковзного вікна фіксованого розміру. Теоретичні результати обґрунтовано та підтверджено моделюванням. Розроблено імітаційну модель, що враховує вплив ймовірності бітової помилки в каналі зв'язку, дозволяючи оцінити ймовірність правильних і хибних спрацювань підсистеми синхронізації. Побудована імітаційна модель дозволила визначити розподіл довжин серій хибних спрацювань підсистеми синхронізації для ймовірності бітової помилки в каналі зв'язку 0.4 та довжини перестановки-синхрокомбінації $M = 8$. Визначене порогове значення довжини серії хибних спрацювань підсистеми синхронізації підвищує ймовірність встановлення правильної кадрової синхронізації за умови невідомого моменту початку приймання синхрокомбінації приймачем і високого рівня шуму в каналі зв'язку. Запропоновані підходи спрямовані на інтеграцію процесів синхронізації та захищеного передавання в одному протоколі інформаційного обміну, забезпечуючи підвищення достовірності та стійкості систем передавання даних.

У роботі виявлено та досліджено сімейства квадратних матриць розміром 2×2 , елементи яких належать простому полю лишків. Показано, що ці матриці формують комутативну (абелеву) групу за операцією множення. Встановлено формулу для обчислення порядку цієї групи, що залежить від фіксованих параметрів. Доведено, що комутативне сімейство матриць одночасно діагоналізується над розширеним полем, утвореним додаванням квадратного кореня з дискримінанта. Це дозволяє розглядати сімейство матриць як поле Галуа порядку p^2 зі стандартними операціями додавання та множення матриць.

Розроблено алгоритми відображення матриць у перестановки через перетворення в десяткову та факторіальну системи числення, що забезпечують рівномірний розподіл перестановок і можуть застосовуватись для генерації ключів на основі матричних структур.

Розроблено апаратний макет системи на базі nRF52840 і ISM-радіоканалу. Проведено порівняльний аналіз ефективності виконання операцій над перестановками на центральному процесорі (CPU) та графічному процесорі (GPU) із застосуванням технології CUDA. Отримані результати засвідчили суттєве скорочення часу обчислень при використанні GPU, що підтверджує доцільність застосування паралельних обчислювальних архітектур для підвищення продуктивності систем захищеного інформаційного обміну, зокрема для прискорення процесів генерації, обробки та перевірки перестановок під час реалізації криптографічних протоколів. Отримані результати ефективності операцій множення над перестановками дозволили розрахувати час проведення атаки грубої сили на трьохетапний протокол з НФКД.

Результати дисертаційної роботи підтверджують ефективність запропонованих рішень як у теоретичному, так і в експериментальному аспектах, створюючи підґрунтя для їх впровадження в практичні системи захищеного інформаційного обміну. Отримані результати цієї роботи формують наукове підґрунтя для подальшої розробки інтегрованих протоколів захищеного інформаційного обміну, що поєднують процеси шифрування, синхронізації та обробки даних у єдиній системі, адаптованій до роботи в умовах непередбачуваних завад та невідомого моменту початку передавання.

Основні результати дисертаційної роботи опубліковано в 11 наукових працях, серед яких: 8 наукових статей (з них 4 – у виданнях, індексованих у Scopus, і 4 – у фахових наукових виданнях України); 2 публікації у збірниках матеріалів міжнародних та всеукраїнських наукових конференцій; 1 підрозділ у колективній науковій монографії.

Ключові слова: перестановка, синхрокомбінація, кадрова синхронізація, імітаційна модель, криптографічний протокол, факторіальне кодування, шум, скінченні поля матриць, імовірність синхронізації, оцінка ймовірності, кореляція, достовірність, надійність, канал зв'язку, кібербезпека.

SUMMARY

Skutskyi A.B. Method and Models of a Secure Information Exchange System with Non-Separable Factorial Data Coding – Qualifying scientific work (manuscript).

Dissertation submitted for the degree of Doctor of Philosophy in specialty 123 – Computer Engineering. – Cherkasy State Technological University, Cherkasy, 2025.

The dissertation is dedicated to the development of a secure information exchange system employing non-separable factorial data coding (NFCD) under noisy communication channel conditions. The increasing volume of confidential data transmitted over open communication channels necessitates improvements in synchronization and encryption methods to prevent unauthorized access and to ensure data integrity and reliability. Synchronization issues become particularly critical when external influences or attacks can cause failures in establishing synchronism, thereby disrupting system performance. Consequently, this work addresses the challenge of organizing secure and reliable information transmission when the initial transmission moment is unknown and communication channels experience high noise levels. To address this challenge, an analysis of existing frame synchronization methods utilizing NFCD was conducted, enabling the identification of further research directions and formulation of the study's objectives.

A mathematical model for detecting synchronization combinations in NFCD-based data transmission systems is presented. Two frame synchronization methods are analyzed: one based on splitting the codeword into prefix and suffix parts, and another based on correlation processing. Both methods employ majority bit processing to enhance reception reliability. The correlation-based frame synchronization method and the method for reliable permutation transmission rely on selecting permutations with the maximal minimum Hamming distance between their binary representations and their cyclic shifts, thereby improving synchronization robustness under high noise intensity. Mathematical models estimating the probabilities of correct and false synchronization for each method are

developed. A method for reliable permutation transmission over noisy channels is described for integration into NFCD-based information exchange and the frame synchronization process.

The dissertation identifies the limitations of existing frame synchronization methods for non-separable factorial codes. In particular, it highlights the problem of an unknown start time for transmitting synchronization sequences in channels with a high bit error probability, which may lead to false triggers or emergency receiver states in the absence of actual transmission. Receiver responses under such conditions are uncertain due to noise influence, increasing the risk of incorrect synchronization establishment.

A review of research on matrix-based cryptographic transformations provided the foundation for integrating matrix structures with NFCD. The study identifies the need to develop approaches for constructing finite fields of 2×2 square matrices and applying them to key-permutation agreement in NFCD systems. It explores modern developments in matrix encryption protocols and key agreement schemes based on mathematical structures in cryptography. The advantages of matrix-based methods are highlighted, particularly the high computational complexity of inverse transformations, which provides resistance to decryption without a key. The role of finite fields as a theoretical basis for cryptographic algorithms is analyzed, including their applications in symmetric encryption, primality testing, factorization, and digital signature generation. While the potential of finite matrix fields in cryptographic schemes is emphasized, the lack of a bijective mapping between matrices and permutations is noted as a challenge for integrating matrix protocols with NFCD. The dissertation underscores the need to develop and study algorithms enabling such mappings. The high potential of matrix-based approaches in cryptography is affirmed, and directions for further theoretical and applied research are outlined.

The object of the study is the process of secure permutation transmission under high-noise conditions in communication channels. The subject of the study is the methods and models of a secure information exchange system utilizing NFCD.

The aim of the research is to ensure secure information exchange in NFCD-based systems over noisy communication channels. Enhanced frame synchronization efficiency under high noise levels is achieved by advancing the mathematical model and detection method for synchronization sequences when the initial reception moment is unknown in a channel with a high bit error probability. Additionally, simulation models of the NFCD-based information exchange system were developed to experimentally validate the theoretical results.

Algorithms for converting text messages into permutations were developed and applied in constructing prototype systems for secure text message exchange via open industrial, scientific, and medical (ISM) band communication channels. The dissertation presents the structure of these prototype secure text exchange systems based on NFCD, including their hardware components. The developed prototypes can serve as educational platforms or foundational designs for Internet of Things (IoT) applications.

The scientific novelty of the dissertation lies in further developing the frame synchronization method, which employs majority and correlation processing of received fragments of synchronization sequences represented as permutations. This development builds upon an enhanced mathematical model for detecting synchronization sequences, allowing estimation of correct and false synchronization probabilities using majority and correlation processing within a fixed-size sliding window. Theoretical results were substantiated and validated through simulation. A simulation model was created that accounts for bit error probability in the communication channel, enabling estimation of correct and false synchronization trigger probabilities. The simulation model facilitated determining the distribution of false trigger sequence lengths for a bit error probability of 0.4 and a specified permutation-synchronization sequence length $M = 8$. A threshold value for false trigger sequence length was established, increasing the probability of correct frame synchronization when the initial reception moment is unknown and noise levels are high. The proposed approaches aim to integrate synchronization and secure

transmission processes into a unified information exchange protocol, thereby improving reliability and robustness of data transmission systems.

The study identified and investigated families of 2×2 square matrices over a prime residue field, demonstrating that these matrices form a commutative (abelian) group under multiplication. A formula for calculating the group order based on fixed parameters was derived. It was proven that the commutative matrix family is simultaneously diagonalizable over an extended field formed by adjoining the square root of the discriminant. This enables interpreting the matrix family as a Galois field of order p^2 with standard matrix addition and multiplication operations.

Algorithms for mapping matrices to permutations via conversion into decimal and factorial numeral systems were developed, providing uniform permutation distributions suitable for matrix-based key generation.

A hardware prototype system was developed using an nRF52840 microcontroller and an ISM radio channel. A comparative analysis of permutation operation performance on a central processing unit (CPU) and a graphics processing unit (GPU) using CUDA technology was conducted. The results showed significant computational acceleration on the GPU, confirming the feasibility of parallel computing architectures to enhance the performance of secure information exchange systems, particularly in accelerating permutation generation, processing, and verification for cryptographic protocols. The results also enabled estimating the time required for a brute-force attack on a three-pass NFCD protocol.

The findings of this dissertation confirm the effectiveness of the proposed solutions in both theoretical and experimental aspects, providing a foundation for their implementation in practical secure information exchange systems. The results establish a scientific basis for further development of integrated secure information exchange protocols combining encryption, synchronization, and data processing into a unified system adapted to operate under unpredictable interference and unknown transmission start times.

The main results of the dissertation have been published in 11 scientific works, including: 8 scientific articles (4 of which are published in journals indexed

in Scopus, and 4 in peer-reviewed scientific journals of Ukraine); 2 papers in the proceedings of international and national scientific conferences; 1 book chapter in a collective scientific monograph.

Keywords: permutation, synchronization sequence, frame synchronization, simulation model, cryptographic protocol, factorial coding, noise, finite matrix fields, synchronization probability, probability estimation, correlation, trustworthiness, reliability, communication channel, cybersecurity.

Список публікацій здобувача

- [1] E. Faure, A. Baikenov, A. Skutskyi, D. Faure, i O. Abramkina, «Algorithms for reliable permutation transmission protocols in noisy communication channels», *CEUR Workshop Proceedings*, т. 3826, с. 40-49, 2024, doi: [10.5281/zenodo.15390412](https://zenodo.org/doi/10.5281/zenodo.15390412) (Scopus)
- [2] E. Faure, A. Shcherba, A. Skutskyi, i A. Lavdanskyi, «A Finite Field of Square Matrices of Order 2», *CEUR Workshop Proceedings*, т. 3550, с. 306-312, 2023, doi: [10.5281/zenodo.15392022](https://zenodo.org/doi/10.5281/zenodo.15392022) (Scopus)
- [3] E. Faure, A. Shcherba, A. Skutskyi, i A. Lavdanskyi, «A software model to generate permutation keys through a square matrix», *Вісник Черкаського державного технологічного університету*, т. 29, № 2, с. 10-23, 2024, doi: [10.62660/bcstu/2.2024.10](https://bcstu.org.ua/doi/10.62660/bcstu/2.2024.10)
- [4] E. Faure, A. Skutskyi, i A. Lavdanskyi, «Algorithms and simulation model for the synchronisation subsystem of the noise-resilient communication system based on permutations», *Вісник Черкаського державного технологічного університету*, т. 4, № 29, с. 62-74, 2024, doi: [10.62660/bcstu/4.2024.62](https://bcstu.org.ua/doi/10.62660/bcstu/4.2024.62)
- [5] A. Lavdanskyi, E. Faure, A. Skutskyi, i C. Bazilo, «Accelerating Operations on Permutations Using Graphics Processing Units», *Lecture Notes on Data Engineering and Communications Technologie*, т. 178, с. 3-12, 2023, doi: [10.1007/978-3-031-35467-0_1](https://doi.org/10.1007/978-3-031-35467-0_1) (Scopus)

- [6] A. Shcherba, E. Faure, A. Skutskyi, i O. Kharin, «Families of Square Commutative 2x2 Matrices», *CEUR Workshop Proceedings*, т. 3550, с. 289-296, 2023, doi: [10.5281/zenodo.15391901](https://zenodo.org/record/15391901) (Scopus)
- [7] А. О. Лавданський, Е.В. Фауре, С. Т. Тинимбаєв, і А. Б. Скуцький, «Система захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону», *Вісник Черкаського державного технологічного університету*, т. 27, № 3, с. 41-48, 2022. doi: [10.24025/2306-4412.3.2022.267786](https://doi.org/10.24025/2306-4412.3.2022.267786)
- [8] Е. В. Фауре, А. Б. Скуцький, і А. О. Лавданський, «Імітаційна модель передавання текстових і аудіо повідомлень з використанням нероздільного факторіального кодування в середовищі Simulink», в *Challenges and threats to critical infrastructure*, Detroit, Michigan, USA: NGO Institute for Cyberspace Research, 2023, с. 244-246. [Online]. Режим доступу: <https://er.chdtu.edu.ua/bitstream/ChSTU/4539/1/Monograph-09-06-2023-Faure2.pdf>
- [9] Е. В. Фауре, А. Б. Скуцький, і А. О. Лавданський, «Імітаційна модель системи передавання інформації з нероздільним факторіальним кодуванням даних у середовищі Simulink», *Вісник Черкаського державного технологічного університету*, т. 27, № 4, с. 31-47, 2022, doi: [10.24025/2306-4412.4.2022.273385](https://doi.org/10.24025/2306-4412.4.2022.273385)
- [10] Е. В. Фауре і А. Б. Скуцький, «Розробка моделі трьохетапного криптографічного протоколу на основі перестановок», в *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей XII Міжнародної науково-технічної конференції, Баку–Харків–Жиліна, 27–28 квітня 2022 року*, Харків: ХНУРЕ, 2022, с. 138. [Online]. Режим доступу: https://nure.ua/wp-content/uploads/conf-2022-akov/telecom_2022_volume_1.pdf
- [11] Е. В. Фауре, А. Б. Скуцький, А. О. Лавданський, і О. О. Харін, «Протокол надійного передавання перестановок в умовах інтенсивних шумів у каналі зв'язку» в *Інновації та перспективні шляхи розвитку*

інформаційних технологій (ІПШРІТ-2024): тези доповідей III Міжнародної науково-практичної інтернет-конференції, Черкаси: ЧДТУ, 2024, с. 107. [Online]. Режим доступу : https://drive.google.com/file/d/15-8DffQpER_5F6TniHYNIDf2BjOPjehX/view?usp=drive_link

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	18
ВСТУП	19
1. АНАЛІЗ СУЧАСНОГО СТАНУ ПРЕДМЕТНОЇ ОБЛАСТІ. ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ	30
1.1. Вступ.....	30
1.2. Нероздільне факторіальне кодування даних	32
1.3. Математична модель процесу виявлення синхрокомбінації систем передавання інформації з нероздільним факторіальним кодуванням даних..	35
1.3.1. Метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі поділу кодового слова на префіксну й суфіксну частини	38
1.3.2. Метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі кореляційної обробки ..	42
1.3.3. Надійне передавання перестановок в каналах з високою інтенсивністю бітової помилки	46
1.4. Матричні протоколи шифрування та узгодження ключів	49
1.5. Цілі та задачі дисертаційного дослідження.....	54
1.6. Висновки	56
2. МЕТОД КАДРОВОЇ СИНХРОНІЗАЦІЇ СИСТЕМ ПЕРЕДАВАННЯ ДАНИХ З НЕРОЗДІЛЬНИМ ФАКТОРІАЛЬНИМ КОДУВАННЯМ	58
2.1. Вступ.....	58
2.2. Схема кадрової синхронізації з використанням ковзного вікна	59
2.3. Оцінки ймовірнісних показників кадрової синхронізації.....	61
2.4. Підхід до зменшення ймовірності хибної синхронізації	69
2.5. Підхід до забезпечення ефективної синхронізації за відомою максимальною довжиною серії хибних спрацювань.....	73
2.6. Опис методу	74

	15
2.7. Алгоритми та імітаційна модель підсистеми кадрової синхронізації ..	75
2.7.1. Процедура інформаційного обміну та схема його реалізації	76
2.7.2. Структура блоку синхронізації.....	78
2.7.3. Довжина серії хибних спрацювань	79
2.7.4. Імітаційна модель системи інформаційного обміну на основі перестановок.....	83
2.7.5. Алгоритми роботи компонентів системи інформаційного обміну на основі перестановок	86
2.7.6. Результати роботи імітаційної моделі	91
2.7.7. Особливості імітаційної моделі.....	95
2.8. Висновки	100
3. СКІНЧЕННІ ПОЛЯ КВАДРАТНИХ МАТРИЦЬ ПОРЯДКУ 2.....	102
3.1. Вступ.....	102
3.2. Комутативні сімейства матриць 2×2	102
3.3. Комутативне сімейство матриць 2×2 з одиницею	107
3.4. Діагоналізація матриць групи $CGL_{b,k}(2, \mathbb{Z}_p)$	111
3.5. Програмна модель формування ключів-перестановок через квадратну матрицю	117
3.5.1. Опис методу формування перестановки з квадратної матриці....	118
3.6. Програмна модель перетворення матриць другого порядку в перестановки та алгоритми роботи	119
3.6.1. Алгоритм роботи програмної моделі перебору матриць.....	122
3.6.2. Особливості програмної моделі	123
3.6.3. Збір статистичних даних і їх валідація	124
3.6.4. Оцінка за критерієм Пірсона χ^2 результатів формування ключів-перестановок через квадратну матрицю.....	137
3.7. Висновки	141

4. ПРАКТИЧНІ АСПЕКТИ РЕАЛІЗАЦІЇ СИСТЕМ ІНФОРМАЦІЙНОГО ОБМІНУ З НЕРОЗДІЛЬНИМ ФАКТОРІАЛЬНИМ КОДУВАННЯМ ДАНИХ.....	143
4.1. Вступ.....	143
4.2. Імітаційна модель системи інформаційного обміну	144
4.2.1. Структура імітаційної моделі	144
4.2.2. Налаштування імітаційної моделі	148
4.2.3. Підсистеми імітаційної моделі	150
4.2.4. Опис алгоритму підрахунку перестановок, не пошкоджених помилкою	159
4.2.5. Підсистема фіксації перестановок з невиявленою помилкою.	159
4.2.6. Результати роботи імітаційної моделі інформаційного обміну перестановками	160
4.2.7. Оцінка роботи імітаційної моделі	166
4.3. Побудова прототипу захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону	168
4.3.1. Алгоритм кодування текстових повідомлень в перестановку	168
4.3.2. Приклад перетворення послідовності текстових символів у перестановку	170
4.3.3. Алгоритм приймання та декодування перестановок у текст	171
4.3.4. Приклад декодування перестановки у текст	172
4.3.5. Структура і опис прототипу системи захищеного інформаційного обміну текстовими повідомленнями ISM радіоканалом.....	173
4.3.6. Особливості макетних зразків	177
4.4. Використання графічних прискорювачів для операцій над перестановками	178
4.4.1. Особливості архітектури графічних процесорів.....	179
4.4.2. Застосування графічних процесорів у криптографічних перетвореннях.....	181

4.4.3. Оцінка швидкодії виконання операцій над перестановками при використанні графічних процесорів.....	182
4.4.4. Оцінка результатів використання графічних процесорів для операцій над перестановками	185
4.5. Висновки	189
ВИСНОВКИ.....	191
СПИСОК ДЖЕРЕЛ	195
ДОДАТКИ.....	218
Додаток А. Лістинги розрахунково-експериментальних моделей	218
А.1. Лістинг імітаційно-програмної моделі інформаційного обміну з НФКД на мові Matlab.....	218
А.2. Лістинг імітаційно-програмної моделі визначення величини серії хибних спрацювань підсистеми синхронізації з НФКД на мові Matlab	224
А.3. Лістинг імітаційно-програмної моделі перетворення квадратної матриці другого порядку в перестановку на мові Matlab	229
Додаток Б.Список публікацій здобувача за темою дисертації та відомості про апробацію результатів дисертації.....	235

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

AES – Advanced Encryption Standard;

ASCII – American Standard Code for Information Interchange;

BSC – Binary Symmetric Channel;

CUDA – Compute Unified Device Architecture;

DES – Data Encryption Standard;

DSS – Digital Signature Standard;

FPGA – Field-Programmable Gate Array;

IEEE – Institute of Electrical and Electronics Engineers;

ISM – Industrial, Scientific and Medical;

IoT – Internet of Things

NIST – National Institute of Standards and Technology;

RSA – Rivest–Shamir–Adleman;

НФКД – нероздільне факторіальне кодування даних;

ФКВД – факторіальне кодування з відновленням даних за перестановкою;

ВСТУП

Актуальність теми дослідження. Сучасна інформаційна безпека є одним із ключових аспектів розвитку суспільства у XXI столітті. З поширенням цифрових технологій, зростанням обсягів переданих даних та розвитком глобальних комунікаційних мереж захист інформації стає критично важливим. У відкритих або зашумлених каналах передавання даних [1], таких як Інтернет чи мобільні мережі, виникають серйозні ризики втрати конфіденційності, цілісності та доступності інформації.

Нові підходи в створенні криптографічних систем спрямовані на інтеграцію квантових технологій [2], [3], [4], адаптацію до середовищ із відкритими каналами передавання [5], а також на розробку алгоритмів [6], [7], які здатні протистояти загрозам, що виникають у результаті зростання обчислювальних потужностей комп'ютерних систем та квантових обчислень [8]. Одним із перспективних напрямів розробки новітніх алгоритмів, що мають потенціал протистояти сучасним і майбутнім загрозам, є побудова криптографічних протоколів, що базуються на матричних обчисленнях [9], [10], [11], [12] і нероздільному факторіальному кодуванні даних (НФКД) [13], [14], [15], [16], [17].

Матричні протоколи поєднують високу обчислювальну складність із ефективністю реалізації, що робить їх перспективними для передавання конфіденційної інформації, організації захисту в базах даних, створення стійких систем захисту банківських транзакцій, побудови безпечних каналів зв'язку, систем електронної комерції та голосування [10], [11], [18], [19], [20]. Сучасні матричні криптографічні протоколи демонструють широкий спектр застосувань алгебраїчних структур для побудови надійних методів шифрування та узгодження ключів. Зокрема, перспективним є використання матричних аналогів класичного протоколу Діффі-Хеллмана, заснованих на матрицях Галуа та матрицях Фібоначчі [21]. Іншим прикладом є протоколи, що ґрунтуються на матричних перестановках значної розмірності, які забезпечують підвищену криптографічну стійкість завдяки складності

матричних операцій [22]. Також у сучасних дослідженнях розглядаються узагальнені матриці Фібоначчі для реалізації шифрів типу Affine-Hill, що дозволяє зменшити обчислювальну складність та оптимізувати передачу даних [23]. Незважаючи на значні досягнення у цій сфері, на сьогодні практично відсутні дослідження, присвячені обґрунтуванню принципів і методів побудови скінченних полів квадратних матриць над скінченними полями простих чисел. Застосування скінченних полів квадратних матриць має перспективи для розвитку компактних і ефективних матричних протоколів з високим рівнем криптографічної стійкості.

У свою чергу, використання НФКД [13], [14], [24], [25], [26] відкриває нові можливості для захисту інформації в умовах середовища з високим рівнем шуму. Такі коди дозволяють, як виявляти, так і ефективно виправляти помилки, що виникають у каналі зв'язку завдяки надлишковості, притаманній представленню даних у вигляді перестановок.

У системах із застосуванням НФКД особливої актуальності набуває задача встановлення синхронізації кадрів [27]. Коли межі інформаційного блоку визначаються некоректно через спотворення чи зсув внаслідок дії завад, відбувається повна втрата змісту повідомлення [25], [28]. Постає задача забезпечення синхронізації в умовах високої інтенсивності шуму в каналі зв'язку за невідомому початковому моменті приймання. Такі ситуації становлять суттєву загрозу, зокрема в системах реального часу, де критично важливим є не лише захист даних, а й їх своєчасна доставка. Вирішення цієї задачі полягає в тому, щоб точно визначити момент, коли у вхідному потоці з'являються корисні дані від передавача. За наявності шуму процедура декодування та встановлення синхронізації стає складною у вирішенні. Вплив завад у каналі на інформаційний потік може призвести до помилок або аварійних станів на етапі декодування інформації. У випадку застосування НФКД, де одна помилка в процесі обміну може порушити декодування всього інформаційного потоку [29], це особливо важливо. Тому правильне встановлення кадрової синхронізму за невідомого початкового моменту

передавання синхрокомбінацій у каналі з високою імовірністю бітової помилки є критично важливим елементом.

Розвиток систем передавання даних на основі НФКД потребує комплексного підходу – від побудови математичних моделей і верифікації їх на імітаційному рівні, до створення макетних зразків, здатних працювати в умовах зашумлених каналів. Підхід з використанням програмних та імітаційних моделей допомагає у вирішенні багатьох задач, зокрема і під час побудови системи інформаційного обміну. Введення абстракції допомагає зосередитись на головних моментах і проблемах дослідження. При досягненні обмежень певної абстракції моделі є можливість доповнити та розширити її, тим самим ефективніше проводити розробку, перевірку ефективності та тестування нових методів і підходів. Важливою складовою є оцінка обчислювальної ефективності алгоритмів кодування та декодування, а також можливості їх перевірки з використанням графічних процесорів (GPU) або апаратних рішень на програмованих логічних матрицях (FPGA) [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], що відкриває шлях до практичного впровадження у системах Інтернету речей (IoT) [32], [36], [40], [41], [42], [43], [44], [45].

Питанням підвищення ефективності кодування інформації і пошуком нових кодувальних схем займалися такі вчені, як R.J. McEliece [46], F.J. MacWilliams та N.J.A. Sloane [47], О.А. Борисенко [48], [49], [50], І.Д. Горбенко [51], [52], В.І. Грабчак [53], [54], [55], О.О. Кузнецов [56], [57], [58], [59], Е.Л. Онанченко [60], [61], О.П. Стахов [62], [63], [64], [65], В.Я. Чечельницький [66], [67], [68], [69]. Дослідженням впливу шуму в системах інформаційного обміну присвячені праці таких дослідників, як J. Massey [70], M. Chiani та M. Martini [71], M. Hasler та T. Schimming [72], О.О. Кузнецов [56]. Дослідженням побудови захищених систем інформаційного обміну, зокрема з використанням перестановок і факторіального кодування, займались такі науковці, як T. Berger [73], D. H. Smith [74], P. J. Cameron [75], I. F. Blake [76], V. K. Goyal [77], О.А. Борисенко [48], [78], [79], [80], [81], [82], [83], [84], Е.В. Фауре [14],

[29], [85], [86], [87], [88], [89], [90], [91], [92], [93], [94], [95], А.І. Щерба [14], [29], [85], [86], [87], [89], [89], [90], [95], [96], [97], [98], [99], [100], [101], [102], [103], А.О. Лавданський [93], [104], [105], Б.А. Ступка [16], [85], [90], [91], [92], [99], [101], В.В. Швидкий [89], [90], [97], О.О Харін [26], [93], [106].

Питанням використання матричних структур і розвитку теорії матричного кодування присвячено роботи дослідників А. Нock [107], D. Serre [108], D. Bigatti [109], W. P. Wardlaw [110], L. Brickman [111], R. Bellman [112], R.J. McEliece [46], [113], M.K. Singh [114], О.П. Стахов [63], [64]. До сучасних розробок криптографічних схем кодування, що базуються на узагальнених матрицях Фібоначчі, матричних перестановках і полях Галуа належать роботи авторів А. Naseri [10], M. Durcheva [11], M. Maxrizal [12], M. Abu-Faraj [18], F. Al-Shaarani і A. Gutub [19], T. Kumar і S. Chauhan [20], А. Білецький [21]. Однак у зазначених роботах підходи до забезпечення криптографічної стійкості фокусуються на загальних методах матричної криптографії. Перспективним же напрямом є розробка підходів узгодження ключів на основі скінченних полів квадратних матриць та інтеграція з НФКД, що залишається малодослідженими у сучасній літературі й становлять актуальне наукове завдання.

Виходячи з цього, тема дисертаційної роботи «Метод та моделі системи захищеного інформаційного обміну з нероздільним факторіальним кодуванням даних» є актуальною.

Зв'язок роботи з науковими програмами, планами, темами.

Дослідження, результати яких представлено в дисертаційній роботі, відповідають пріоритетному напрямку розвитку науки і техніки України «Інформаційні та комунікаційні технології» та його тематичному напрямку «Інформаційно-комунікаційні та радіоелектронні системи та технології, засоби радіоелектронної боротьби для забезпечення національної безпеки і оборони. Інформаційна безпека та кібербезпека» і виконувалися відповідно до програм і планів науково-дослідних робіт Черкаського державного технологічного університету, в тому числі в рамках держбюджетних науково-

технічних (експериментальних) розробок молодих вчених, де автор був виконавцем:

- «Розробка завадозахищеної енергоефективної системи контролю та управління віддаленими безпілотними об'єктами на основі факторіального кодування даних» (№0125U000637);
- «Розробка методів, протоколів і засобів захищеного інформаційного обміну з використанням трьохетапного криптографічного протоколу на основі перестановок в умовах зашумленості каналів зв'язку» (№0123U100270);
- «Розробка мобільної системи захищеного інформаційного обміну для військових і цивільних підрозділів державних структур» (№0120U102607).

Метою дослідження є забезпечення захищеного інформаційного обміну в системах з нероздільним факторіальним кодуванням даних і зашумленим каналом зв'язку.

Виходячи з зазначеної мети, задачами роботи є:

- розвинути метод кадрової синхронізації нероздільних факторіальних кодів для випадків невідомого моменту початку приймання синхрокомбінацій передавача;
- розвинути математичну модель процесу виявлення синхрокомбінації систем передавання інформації з нероздільним факторіальним кодуванням даних за невідомого моменту початку приймання синхрокомбінацій передавача;
- розробити математичну модель скінченного поля квадратних матриць порядку 2 над скінченим полем простих чисел та розробити методику їх використання для узгодження ключів-перестановок;
- побудувати імітаційну модель системи інформаційного обміну, провести порівняльні експериментальні оцінки удосконаленого методу кадрової синхронізації.

Об'єктом дослідження є процес захищеного передавання перестановок в умовах високої інтенсивності шуму в каналі зв'язку.

Предметом дослідження є метод і моделі системи захищеного інформаційного обміну з нероздільним факторіальним кодуванням даних.

Методи дослідження. У дисертаційній роботі використано сукупність загальнонаукових і спеціальних методів. До спеціальних методів віднесено: методи теорії ймовірностей та математичної статистики; методи дискретної математики; методи теорії скінченних полів і лінійної алгебри; методи комп'ютерного моделювання; методи експериментального дослідження.

Наукова новизна отриманих результатів:

– *набув подальшого розвитку* метод кадрової синхронізації нероздільних факторіальних кодів, що використовує як синхрокомбінацію перестановку чисел, яка має максимальне значення мінімальної відстані Хеммінга від її двійкового представлення до всіх її циклічних зсувів, а також кореляційну та мажоритарну обробку прийнятих фрагментів, де довжина фрагмента дорівнює довжині синхрокомбінації, який за рахунок використання ковзного вікна фіксованого розміру та врахування серії спрацювань підсистеми синхронізації дозволяє встановити кадрову синхронізацію приймальної та передавальної станцій системи інформаційного обміну за високої ймовірності бітової помилки та невідомого моменту початку приймання синхрокомбінацій передавача, забезпечуючи необхідні показники ймовірності правильної та хибної синхронізації;

– *набула подальшого розвитку* математична модель процесу виявлення синхрокомбінації систем передавання інформації з нероздільним факторіальним кодуванням даних, що використовують кореляційну та мажоритарну обробку прийнятих фрагментів, яка за рахунок дослідження механізмів перетворення синхрокомбінації в її зсув в умовах застосування приймачем ковзного вікна фіксованого розміру дозволяє оцінити ймовірності правильної та хибної кадрової синхронізації.

– *вперше розроблено* математичну модель скінченного поля квадратних матриць порядку 2 над скінченним полем простих чисел \mathbb{Z}_p , яка за рахунок збільшення порядку поля квадратних матриць до значення p^2 , проте збереження порядку p поля \mathbb{Z}_p , в якому виконують операції перетворення, дозволяє підвищити стійкість криптографічних систем, які базуються на операціях у скінченному полі.

Практичне значення результатів дослідження.

1. Розроблено алгоритм кадрової синхронізації на основі ковзного вікна та серії спрацювань підсистеми синхронізації, що може використовуватися для організації захищеного інформаційного обміну у симплексних каналах із високою імовірністю бітових помилок без попереднього узгодження моменту приймання кадрової синхрокомбінації. Розроблений алгоритм забезпечив успішне встановлення кадрової синхронізації в кожному з 1000 випробувань імітаційної моделі за довжини перестановки-синхрокомбінації $M=8$. Кожне випробування передбачало встановлення кадрової синхронізації для ймовірностей бітової помилки в каналі від $p_0=0.1$ до $p_0=0.4$ та процедури надійного передавання блоку корисних даних при відсутності попереднього узгодження початкового моменту інформаційного обміну.

2. Розроблено імітаційну модель системи обміну перестановками, що дозволяє досліджувати вплив різних значень імовірності бітової помилки в каналі зв'язку p_0 і структури синхрокомбінації на імовірнісні показники синхронізації. Отримані результати проведення 10000 випробувань дозволили експериментально визначити максимальну довжину серії хибних спрацювань підсистеми синхронізації $l_{false_synch}=4$ для довжини ковзного вікна 75 фрагментів і $p_0=0.4$. Отримане значення максимальної довжини серії хибних спрацювань підсистеми синхронізації дозволило задати мінімальний поріг серії спрацювань підсистеми синхронізації, тим самим забезпечуючи

необхідні показники ймовірності встановлення правильної кадрової синхронізації.

3. Розроблено алгоритм перетворення матриць у перестановки, що може бути використаний для перетворення квадратних матриць у перестановки при генерації ключів-перестановок та інтегрувати матричні обчислення і НФКД.

4. Створено макетний зразок системи інформаційного обміну, побудований на базі мікроконтролера nRF52840 із використанням ISM-радіоканалу. Макет може бути застосований як прототип для побудови безпечних IoT-рішень або навчальний стенд. Розроблені алгоритми обробки та перетворення текстових повідомлень у перестановки дозволяють стискати текстові повідомлення, що підлягають передаванню в вигляді перестановок. Кількість бітів, що необхідна для передавання двох текстових символів латинського алфавіту (закодовані в перестановку), становить 3 байти замість $M = 8$ байтів, необхідних у звичайному поданні ASCII. Введена надлишковість у текстовому повідомленні забезпечує криптографічний захист, дає можливість перевіряти цілісність інформації шляхом застосування додаткових методів НФКД. Створені приймально-передавальні пристрої побудовані на базі системи на кристалі (System on Chip, SoC) nRF52840 від виробника Nordic Semiconductor і не залежать від архітектури операційної системи комп'ютерної системи, де необхідно реалізувати захищений обмін, можуть бути використані IoT рішеннях.

5. Розроблено програмний код для операцій над перестановками з використанням GPU (CUDA). Розроблені алгоритми множення перестановок дозволили оцінити доцільність використання GPU для прискорення криптографічних операцій над перестановками у системах криптографічного захисту на основі НФКД в порівнянні з обчисленнями на CPU і можуть застосовуватися для оцінки стійкості криптографічних протоколів до атак методом перебору, а також для реалізації високопродуктивних обчислень у серверних системах і системах захищеного інформаційного обміну.

Результати дослідження ефективності GPU обчислень, можуть бути використані під час вибору апаратної платформи для реалізації криптографічних протоколів.

Особистий внесок здобувача. Дисертація є самостійно виконаною завершеною роботою здобувача. Наукові результати і практичні розробки, що містяться в дисертаційній роботі, отримані автором самостійно.

У роботах, опублікованих у співавторстві, автором: [1], [11] – розвинуто математичну модель процесу виявлення синхрокомбінації із застосуванням ковзного вікна; [4] – отримав подальший розвиток метод кадрової синхронізації на основі кореляційної обробки з урахуванням ковзного вікна і серії спрацювань підсистеми синхронізації, досліджено ефективність застосування розробленого методу синхронізації в симплексному каналі передавання з імовірністю бітової помилки 0.4; [2], [6] – теоретично обґрунтовано принципи побудови скінченного поля квадратних матриць порядку 2 для криптографічних застосувань, визначено шість сімейств матриць із загальної лінійної групи $GL(2, \mathbb{Z}_p)$ порядку 2 над простим полем цілих чисел за модулем p з комутативною операцією множення; [3] – розроблено та досліджено алгоритми перетворення квадратних матриць порядку 2 в перестановки, наведені статистичні властивості отриманих результатів перетворення, визначено найбільш ефективний алгоритм перетворення матриць у перестановки за критерієм рівномірності розподілу отриманих номерів перестановок; [5] – дослідження підвищення ефективності виконання операцій над перестановками за допомогою графічних процесорів (GPU) в порівнянні з центральним процесором (CPU), перевірено стійкість трьохетапного протоколу на перестановках до атаки методом перебору ключових перестановок; [7] – створено систему захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону на основі НФКД, наведено опис алгоритмів перетворення тексту в перестановки і навпаки; [8], [9] – розроблено та досліджено імітаційні моделі що використовують канал зв'язку з незалежними бітовими помилками для передачі текстової або аудіо

інформації у вигляді перестановок; [10] – створено імітаційну модель трьохетапного криптографічного протоколу на основі перестановок.

З робіт, опублікованих у співавторстві, для вирішення задач, поставлених у дисертаційному дослідженні, використано результати, отримані здобувачем особисто.

Апробація результатів дисертації. Основні результати дисертаційної роботи доповідалися та обговорювалися на:

- XII Міжнародній науково-технічній конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» (Баку–Харків–Жиліна, 27–28 квітня 2022 року);
- Міжнародна науково-практична конференція «Information Technology for Education, Science and Technics» (ITEST 2022);
- II Міжнародна науково-практична конференція «Виклики та загрози об’єктам критичної інфраструктури» (Київ, 29-30 червня, 2023);
- Cybersecurity Providing in Information and Telecommunication Systems II (CPITS-II-2023), (Kyiv, 26 October 2023);
- Cybersecurity Providing in Information and Telecommunication Systems II (CPITS-II 2024), (Kyiv, 26 October 2024);
- III Міжнародній науково-практичній інтернет-конференції «Інновації та перспективні шляхи розвитку інформаційних технологій» (ІПШРІТ-2024), (Черкаси, 22 листопада 2024 року).

Публікації. Основні результати дисертаційної роботи опубліковано в 11 наукових працях, серед яких: 8 наукових статей (з них 4 – у виданнях, індексованих у Scopus [1], [2], [5], [6], і 4 – у фахових наукових виданнях України [3], [4], [7], [9]); 2 публікації у збірниках матеріалів міжнародних та всеукраїнських наукових конференцій [10], [11]; 1 підрозділ у колективній науковій монографії [8].

Структура і обсяг дисертаційної роботи. Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних

джерел і додатків. Загальний обсяг роботи становить 237 сторінок машинописного тексту, включає 62 рисунка, 11 таблиць та 197 найменувань у списку використаних джерел.

У **першому розділі** обґрунтовано актуальність теми, сформульовано мету, задачі, об'єкт і предмет дослідження, а також подано аналіз стану предметної області.

У **другому розділі** розроблено метод кадрової синхронізації на основі ковзного вікна та аналізу серії спрацювань підсистеми синхронізації для систем інформаційного обміну з використанням нероздільного факторіального кодування в симплексному каналі зв'язку. Побудовано математичну модель процесу приймання перестановок у зашумленому середовищі та досліджено ймовірнісні характеристики правильної та хибної синхронізації при застосуванні ковзного вікна.

У **третьому розділі** математично обґрунтовано побудову скінченних полів квадратних матриць порядку 2 у криптографії, розроблено алгоритми формування ключів-перестановок, алгоритми перетворення матриць у перестановки та способи їх представлення у факторіальній системі числення.

У **четвертому розділі** наведено практичну реалізацію системи обміну перестановками, описано архітектуру макетного зразка, алгоритми кодування та декодування текстових повідомлень, порівняно ефективність виконання операцій над перестановками на CPU і GPU.

У **висновках** узагальнено результати дослідження, сформульовано основні наукові положення та визначено перспективи подальших досліджень.

1. АНАЛІЗ СУЧАСНОГО СТАНУ ПРЕДМЕТНОЇ ОБЛАСТІ. ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1. Вступ

Відповідно до звіту [115] Державної служби спеціального зв'язку та захисту інформації України про роботу системи виявлення вразливостей і реагування на кіберінциденти та кібератаки [116], від початку війни тренд на зростання кількості кібератак зберігається. Зокрема, в III кварталі 2022 року за допомогою засобів Системи виявлення вразливостей і реагування на кіберінциденти та кібератаки опрацьовано 24 млрд подій. Кількість зареєстрованих і опрацьованих кіберінцидентів зросла від 64 до 115. Найпоширенішими методами кібератаки є: збір інформації зловмисником, шкідливий програмний код, втручання, порушення доступності [117], [118], [119]. Розвиток сучасних комп'ютерів і технологій дозволяє виконувати кібератаки на смартфоні через спеціалізоване програмне забезпечення [120] та використовувати сучасні апаратні рішення, що є енергоефективними і малогабаритними.

Таке зростання складності та масштабності атак вимагає не лише вдосконалення традиційних засобів захисту, а й пошуку принципово нових підходів до забезпечення інформаційної безпеки. Одним із перспективних напрямів, що стрімко розвивається, є використання квантових обчислень, здатних кардинально змінити як методи захисту, так і методи атак. У роботі [121] автори пропонують масштабовану паралельну схему для над швидкої генерації випадкових бітів зі швидкістю 100 терабіт на секунду на основі одного мікрокільцевого резонатора, що може застосовуватись як генератор ключів криптографічних протоколів. У роботі [122] з метою дослідження алгоритму Гровера, який є одним із методів у вирішенні задачі криптографії, авторами використано можливості хмарних обчислень на квантових комп'ютерах компанії IBM: *ibmq_16_mellbourne*, *ibmq_london* та інші. У роботі [123] розкривається дослідження того, як квантові обчислення можуть

зламати алгоритм шифрування RSA за поліноміальний час, чого не можуть зробити звичайні комп'ютери через обмеження обчислювальної потужності.

Новітні розробки технології виготовлення напівпровідникових кристалів дозволяють розробляти нові архітектури сучасних графічних процесорів та покращувати їх технічні характеристики. Зокрема використання технології CUDA вже на 2021 рік дозволяє виконувати успішні атаки грубої сили шляхом підбору паролів довжиною 8 елементів з літер нижнього та верхнього регістру, спеціальних символів і цифр всього за 8 годин [124]. У разі збільшення довжини паролів, з метою підвищення стійкості, на користувача накладаються вимоги, що до збереження такого паролю не просто «в голові», а на носіях або в хмарних сховищах. Відповідно такі рішення теж вимагають комплексного підходу при організації захисту, шифрування, обміну та належного зберігання.

Теорія нероздільного факторіального кодування даних [15], [13] дозволяє використовувати перестановки як транспортний механізм у системах зв'язку з короткими пакетами [5], [125], [126], а також реалізувати спільний захист даних від помилок каналу зв'язку і несанкціонованого доступу [127]. Особливістю таких кодів є надлишковість інформації, що передається за допомогою перестановок. Завдяки цьому нероздільні факторіальні коди, як і перестановочні коди (permutation codes) здатні не лише виявляти, але й ефективно виправляти помилки, що виникають у каналі зв'язку. Крім того, специфічна структура перестановок створює сприятливі умови для синхронізації кадрів без необхідності додаткових сигналів, що спрощує роботу системи передавання. За цих умов, довжина синхрослова дорівнює довжині перестановки, що гарантує узгодженість сигналу між передавачем і приймачем.

Тенденції останніх досліджень у криптографії свідчать про подальший розвиток застосування теорії матриць для представлення та перетворення інформації. Використання матричних методів при побудові протоколів шифрування базується на математичних операціях із матрицями, що

забезпечують високий рівень захисту даних через складність їх аналізу та декодування. Значну кількість криптографічних перетворень реалізовано через представлення інформації у вигляді матриць. Зокрема, такий підхід уже підтвердив свою ефективність у стандарті шифрування AES [128].

Враховуючи зростаючу роль матричних перетворень у сучасних криптографічних системах та їхню ефективність при реалізації складних схем шифрування, доцільним є проведення ґрунтовного аналізу наявних протоколів, що базуються на використанні матриць для забезпечення конфіденційності та узгодження криптографічних ключів. Такий аналіз дозволить не лише класифікувати наявні рішення за характером використання матричних структур, а й оцінити їх потенціал для подальшої інтеграції з методами нероздільного факторіального кодування в умовах сучасних вимог до стійкості та ефективності криптографічного захисту.

У першому розділі для постановки задач дисертаційного дослідження необхідно:

- дослідити математичну модель декодування нероздільного факторіального коду;
- дослідити методи кадрової синхронізації нероздільного факторіального коду;
- дослідити підходи до надійного передавання перестановок у каналах з високою інтенсивністю бітової помилки;
- провести аналіз існуючих матричних протоколів шифрування та їх використання.

1.2. Нероздільне факторіальне кодування даних

Основу напрацювань у контексті розвитку факторіального кодування складають праці О.А. Борисенка, А.Е. Горячева [48], [49], [78], [79], [80], [129], Е.В. Фауре [24], [86], [88], [89], [90], [91], [92], [93], [94], [94], [97], [101], [102], О.О. Харіна [26], [106], [130]. Нероздільне факторіальне кодування передбачає використання перестановок у якості носіїв інформації. Відповідно до [88]

перестановка – це упорядкований набір (послідовність) з M чисел множини $\{0; M - 1\}$ кожне з яких зустрічається в ній тільки один раз. Число M у цьому випадку є довжиною.

Характерною особливістю всіх нероздільних факторіальних кодів є повна заміна інформаційного вектора $A(x)$ довжиною k , на перестановку $\pi_n(x)$ символів із множини $\{0; M - 1\}$. Ця перестановка обирається за умови $M! > 2^k$ і $k = \lfloor \log_2 M! \rfloor$, таким чином гарантувати однозначну відповідність кожному слову джерела лише однієї перестановки [95].

Перевагами використання нероздільного факторіального кодування, представленими у роботі [26], є комплексне вирішення ряду ключових завдань у інформаційних системах, спрямованих на підвищення надійності та безпеки інформаційного обміну. В першу чергу, використання нероздільного факторіального кодування дозволяє суттєво підвищити достовірність передачі інформації за рахунок створення кодових конструкцій, які стійкі до помилок різної кратності, особливо помилок парної кратності, що важко виявляються іншими методами. Це досягається завдяки спеціальним алгоритмам, що доповнюють перестановки додатковими перевірними бітами або використовують комбінації з іншими завадостійкими кодами [94], [95], [131], [132].

Важливою перевагою нероздільних факторіальних кодів є інтегрований захист інформації, який дозволяє одночасно забезпечити стійкість до помилок, захист від несанкціонованого доступу та протидію нав'язуванню хибних даних. Це робить нероздільне факторіальне кодування особливо цінним для систем, які передають конфіденційну або критично важливу інформацію. За рахунок побудови сигнально-кодових конструкцій із застосуванням решіток [93] та високою щільністю розташування сигнальних векторів, факторіальне кодування забезпечує максимальну швидкість передачі даних при заданому рівні достовірності. Це дозволяє ефективно використовувати пропускну здатність каналу зв'язку.

Використання факторіальних кодів дає змогу забезпечити кадрову синхронізацію каналів зв'язку без необхідності використання окремого каналу синхронізації, що суттєво зменшує надлишковість і, відповідно, підвищує ефективність використання ресурсів інформаційних систем. До перспективи використання нероздільного факторіального кодування для встановлення кадрової синхронізації при умовах високої інтенсивності шуму в каналі зв'язку за результатами досліджень [16], [27], [85], [90] можна виокремити наступне:

- методи кадрової синхронізації із застосуванням нероздільного факторіального кодування дозволяють досягти високої достовірності синхронізації навіть за ймовірностей бітової помилки в каналі передавання, близькою до 0.5;
- метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі кореляційної обробки забезпечує високу ймовірність правильної синхронізації $P_{true} \geq 0.9997$ і дуже низьку ймовірність хибної синхронізації $P_{false} < 3 \cdot 10^{-43}$ навіть в екстремальних умовах високого рівня шуму з ймовірністю бітової помилки $p_0 \approx 0.495$.

Таким чином, нероздільне факторіальне кодування виступає універсальним і ефективним рішенням, яке забезпечує високу достовірність, конфіденційність та цілісність інформації з використанням відкритих та зашумлених каналів зв'язку.

Попередні дослідження з використанням нероздільних факторіальних кодів продемонстрували ефективність застосування нероздільного факторіального кодування навіть у ситуаціях, коли ймовірність бітових помилок у каналі зв'язку залишається високою. Зокрема, автором Ступкою Б.А. розроблено алгоритми синхронізації перестановок [27], [85], [90]; алгоритми надійного передавання перестановок у системах зв'язку з короткими кадрами [17], [98]. Водночас розроблені методи синхронізації не досліджувались у ситуаціях коли момент початку передавання синхрокомбінацій є невідомим приймачу. Відомості про подію початок

передавання приймач повинен визначати самостійно і враховувати асинхронність цієї події при інформаційному обміні через зашумлений канал зв'язку. Крім того, інтеграція процедур синхронізації та захищеного передавання в межах одного протоколу станом на 2024 рік досі залишається недостатньо дослідженою.

1.3. Математична модель процесу виявлення синхрокомбінації систем передавання інформації з нероздільним факторіальним кодуванням даних

Синхронізація є одним із ключових етапів функціонування будь-якої системи інформаційного обміну, що забезпечує правильне приймання, розділення та обробку переданих даних. У традиційних системах синхронізація ґрунтується на чіткому визначенні меж кадрів або блоків інформації, що можливо за умови стабільного і слабо зашумленого каналу зв'язку. Проте в умовах високої інтенсивності бітових помилок ця задача значно ускладнюється, особливо в симплексних або односторонніх каналах.

Використання нероздільного факторіального кодування (НФКД) у симплексних системах відкриває нові можливості для підвищення стійкості передавання, проте водночас створює специфічні виклики у контексті синхронізації. Зокрема, інформаційні блоки, що кодуються у вигляді перестановок, не містять чітко виражених кордонів між синхрословами та даними, що потребує розробки спеціалізованих підходів до кадрової синхронізації.

Оскільки сучасні цифрові системи зазвичай представляють та використовують інформацію у двійковому вигляді, постає необхідність вибору ефективного кодування перестановок. Найпоширенішим та найпростішим методом є рівномірне двійкове кодування, за якого кожен елемент (символ) перестановки кодується фіксованою кількістю бітів (Таблиця 1.1), кількість бітів для кодування перестановки довжиною M

елементів визначається як $M \cdot \lceil \log_2 M \rceil$, кількість бітів на елемент перестановки при рівномірному кодуванні визначається виразом $l_r = \lceil \log_2 M \rceil$. Залежно від характеристик каналу зв'язку і специфічних вимог до передавання, також можуть використовуватися альтернативні підходи кодування, спрямовані на оптимізацію окремих параметрів передачі.

Таблиця 1.1 Приклад рівномірного двійкового кодування елементів перестановки

Елемент перестановки	0	1	2	...	6	...	$M - 1$
Рівномірний код	000	001	010	...	110	...	$\text{bin}_{\lceil \log_2 M \rceil}(M - 1)$

Обґрунтування вибору рівномірного двійкового кодування елементів перестановок визначається насамперед його простотою, уніфікованістю та широкою поширеністю у цифрових системах. Завдяки використанню фіксованої кількості бітів $M \cdot \lceil \log_2 M \rceil$, рівномірне кодування дозволяє забезпечити зручну обробку інформації на рівні апаратних засобів та програмного забезпечення, спрощує реалізацію алгоритмів обробки і забезпечує однакову довжину інформаційних блоків, що полегшує процес синхронізації. Додатковою перевагою такого підходу є легкість декодування й виявлення помилок, оскільки однаковий розмір блоків дозволяє ефективно реалізовувати методи кадрової синхронізації. Водночас рівномірне двійкове кодування забезпечує передбачуваність характеристик системи, що особливо важливо при проектуванні алгоритмів із кореляційною обробкою для виявлення синхрокомбінацій у каналах із високим рівнем шуму.

Хоча більш складні методи кодування дозволяють зменшити надлишковість чи оптимізувати інформаційну ємність каналу, вони зазвичай ускладнюють апаратну реалізацію та вимагають додаткових обчислювальних

ресурсів. Це робить їх недоцільними для систем із підвищеними вимогами до надійності, швидкості та простоти реалізації. Таким чином, рівномірне двійкове кодування елементів перестановок є оптимальним вибором для систем захищеного інформаційного обміну з нероздільним факторіальним кодуванням даних, та дозволяє досягнути необхідного балансу між простотою реалізації та ефективністю обчислень.

Для забезпечення ефективності кадрової синхронізації перспективним є застосування спеціального класу перестановок, що мають максимальне значення мінімальної відстані Хеммінга. Однак такими характеристиками володіє далеко не кожна перестановка. Використання саме спеціально підібраних структур перестановок дозволяє ефективно реалізувати алгоритми з кореляційною обробкою, що забезпечують надійне виявлення синхрокомбінацій навіть у каналах зв'язку з підвищеним рівнем шуму та помилок.

Роботи [27], [85], [90] демонструють два методи кадрової синхронізації з використанням НФКД в системах достовірного передавання даних за високої ймовірності бітової помилки:

- метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі поділу кодового слова на префіксну й суфіксну частини [90];
- метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі кореляційної обробки [27], [85].

У двох вищезазначених методах використовується процедура мажоритарної обробки бітів у результаті якої отримується уточнена послідовність R в якій частина бітових помилок виправлена.

1.3.1. Метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі поділу кодового слова на префіксну й суфіксну частини

Метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі поділу кодового слова на префіксну й суфіксну частини використовує визначену структуру перестановки π у якості кодового слова, яка виступає у ролі синхрокомбінації. Така перестановка має довжину M елементів, кожен з яких кодується унікальною двійковою послідовністю. При цьому важливо щоб перший елемент кодувався послідовністю, що складається тільки з нулів, другий – тільки з одиниць, третій символ синхрокомбінації в двійковому записі повинен починатися нулем, а останній елемент повинен закінчуватися двійковою одиницею. Інші елементи перестановки спеціально підбираються так, щоб мати максимальну кількість переходів між нулем і одиницею, що полегшує виявлення синхрокомбінації в зашумленому потоці.

Забезпечення виявлення меж елементів синхрокомбінації покладено на префіксу частину, виявлення меж синхрокомбінацій забезпечує суфіксна частина обраної перестановки. Визначений перелік перестановок в [90], що можуть бути використані, як синхрокомбінації у системах достовірного передавання даних з нероздільним факторіальним кодуванням на основі поділу кодового слова на префіксну й суфіксну частини з довжиною перестановки $M = 8$ наведений в (Таблиця 1.2).

Таблиця 1.2 - Перелік перестановок з довжиною $M=8$ для синхрокомбінації

№	Десятковий запис	№	Десятковий запис
1	(0,7,1,2,4,6,5,3)	7	(0,7,2,6,4,5,1,3)
2	(0,7,1,2,6,4,5,3)	8	(0,7,2,6,4,5,3,1)
3	(0,7,1,3,2,4,6,5)	9	(0,7,3,1,2,4,6,5)

4	(0,7,1,3,2,6,4,5)	10	(0,7,3,1,2,6,4,5)
5	(0,7,2,4,6,5,1,3)	11	(0,7,3,2,4,6,5,1)
6	(0,7,2,4,6,5,3,1)	12	(0,7,3,2,6,4,5,1)

Метод кадрової синхронізації, що базується на поділі кодового слова на префіксну та суфіксну частини, передбачає багаторазову передачу синхрокомбінації з введенням надлишковості для підвищення достовірності прийому в умовах каналу зв'язку з невизначеною ймовірністю бітової помилки.

Процедура встановлення кадрової синхронізації на етапі мажоритарної обробки відповідно [90] полягає в накопиченні непарної кількості фрагментів $l=3,5...l$ синхрокомбінації стороною приймача. В якості синхрокомбінації обрана за певними правилами перестановка π довжиною M представлена у двійковому рівномірному коді і довжиною $n = M \lceil \log_2 M \rceil$. Накопичивши визначену кількість фрагментів, кожен біт уточненої послідовності $R(i)$ визначається за правилом, якщо i -ті біти фрагментів $R_l(i)$ містять більше «одиниць», i -тому біту уточненої послідовності $R(i)$ присвоюється бітове значення «одиниці», в іншому випадку – «нуля»:

$$R(i) = \begin{cases} 1, & \text{if } \sum_{l=1}^l R_l(i) > \frac{l}{2}, \text{ де } i = [1;n]. \\ 0, & \text{else} \end{cases}$$

За результатами цієї обробки формується уточнена послідовність, яка підлягає подальшому аналізу на наявність ознак синхронізації.

Кількість бітових помилок, що може бути виправлена у результуючій уточненій послідовності складає $\left(\frac{l-1}{2} \right) \cdot n$.

Суть методу кадрової синхронізації на основі поділу кодового слова на префіксну й суфіксну частини представлений в роботі [90] полягає в наступному.

- 1) Синхрокомбінацією є перестановка π довжини M символів. Елементи перестановки кодуються рівномірним двійковим кодом таким чином, щоб перший елемент записувався послідовністю, що складається тільки з нулів, а другий – тільки з одиниць. Третій елемент синхрокомбінації в двійковому записі повинен починатися нулем, а останній елемент повинен закінчуватися двійковою одиницею. Решта елементів повинна містити максимальну кількість переходів з одиниці в нуль і навпаки (для мінімізації витрат часу на синхронізацію біт) і не повинна містити комбінації $10 \dots 01 \dots 10$.

$$\underbrace{\hspace{1cm}}_{l_r} \quad \underbrace{\hspace{1cm}}_{l_r}$$
- 2) Під час старту процедури пошуку синхронізму в накопичувач приймача записуються три послідовних фрагмента отриманої з каналу послідовності біт. Довжина кожного з фрагментів дорівнює довжині синхрокомбінації $n = M \cdot l_r = M \lceil \log_2 M \rceil$.
- 3) За прийнятими фрагментами формується уточнена послідовність R . Кожен біт цієї послідовності обчислюється за мажоритарним принципом на основі відповідних біт прийнятих фрагментів. Таким чином, якщо i -ті біти фрагментів містять більше «одиниць», i -тому біту уточненої послідовності присвоюється значення «одиниці», в іншому випадку – «нуля».
- 4) В уточненій послідовності R з урахуванням її циклічного зсуву перевіряється наявність комбінації $10 \dots 01 \dots 10$. Якщо її знайдено, причому вона одна, уточнену послідовність циклічно зсувають таким чином, щоб вона починалась з префіксу $0 \dots 01 \dots 1$.

$$\underbrace{\hspace{1cm}}_{l_r} \quad \underbrace{\hspace{1cm}}_{l_r}$$
- 5) Уточнена послідовність R порівнюється з еталоном π . Якщо вона збігається з етальною синхрокомбінацією з точністю до одного з її елементів, то процедура підстроювання циклової фази припиняється, циклічним зсувом компенсується фазова неузгодженість і формується сигнал «Пошук циклової фази завершений». Цей сигнал відправляється

на станцію передавання даних і є командою на перехід до процедури передавання даних користувача.

- б) Якщо комбінацію $10 \dots 01 \dots 10$ в уточненій послідовності не знайдено, l_r l_r знайдено таких комбінацій дві та більше, а також якщо суфікс уточненої послідовності R відрізняється від суфіксу еталонної синхрокомбінації π більше, ніж на один символ, то додатково приймаються ще два фрагменти, тим самим збільшуючи їх число до п'яти. Знову повторюються всі операції виявлення синхрокомбінації, починаючи з пункту 3.
- 7) Число накопичених фрагментів може послідовно збільшуватися до деякого, наперед заданого порогу. Після досягнення цього порогу число накопичених фрагментів не змінюється. Процес пошуку синхронізму триває, поки або не буде знайдений синхронізм, або не закінчиться ліміт часу на виконання пошуку синхронізму. У останньому випадку процедура пошуку синхронізму завершується, а на вихід системи видається сигнал «Аварія каналу».

Відповідно до [90] імовірність бітової помилки в уточненій послідовності R після мажоритарної обробки l прийнятих фрагментів:

$$p_0^* = \sum_{i=\frac{(l+1)}{2}}^l C_l^i p_0^i (1-p_0)^{l-i}. \quad (1.1)$$

Імовірність виникнення бітової помилки в одному елементі перестановки на виході уточненої послідовності оцінюється як:

$$p_{\text{symp}} = \sum_{i=1}^{l_r} C_{l_r}^i (p_0^*)^i (1-p_0^*)^{l_r-i} = 1 - (1-p_0^*)^{l_r}. \quad (1.2)$$

Прийняття рішення про встановлення синхронізації підсистемою синхронізації базується на таких подіях описаних у роботі [90]:

- 1) перші два елементи (двійкове представлення яких містить усі нулі та всі одиниці) уточненої послідовності R визначено без помилки;
- 2) третій елемент уточненої послідовності R починається з двійкового нуля;

- 3) останній елемент уточненої послідовності R закінчується двійковою одиницею;
- 4) не більше одного з групи від третього до останнього елементів уточненої послідовності R визначено невірно.

Імовірнісні показники виникнення зазначених подій:

- імовірність виникнення першої події:

$$P_1 = (1 - p_{\text{symb}})^2 \quad (1.3)$$

- імовірність виникнення 2-4 подій:

$$P_{2-4} = (1 - p_{\text{symb}})^{M-2} + 2 \left((1 - p_0^*) \left(1 - (1 - p_0^*)^{l_r-1} \right) \right) (1 - p_{\text{symb}})^{M-3} + (M - 4) p_{\text{symb}} (1 - p_{\text{symb}})^{M-3} \quad (1.4)$$

Відповідно загальна імовірність встановлення кадрової синхронізації становить:

$$P_{\text{true}} = P_1 P_{2-4}. \quad (1.5)$$

Підставивши відповідні вирази (1.3), (1.4) до (1.5) отримаємо загальний вираз, що оцінює імовірність становлення правильної синхронізації:

$$P_{\text{true}} = (1 - p_{\text{symb}})^{M-1} \left(1 + 2 \left((1 - p_0^*) \left(1 - (1 - p_0^*)^{l_r-1} \right) \right) + (M - 5) p_{\text{symb}} \right).$$

Оцінка імовірності встановлення хибної синхронізації P_{false} детально наведено в роботі [90].

1.3.2. Метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі кореляційної обробки

Недоліками методу кадрової синхронізації на основі поділу кодового слова на префіксну й суфіксну частини є:

- неоптимальність входження в синхронізм за часом;
- висока імовірність хибної синхронізації.

Для усунення зазначених недоліків авторами J. Al-Azzeh, E. Faure, A. Shcherba, і B. Stupka розроблено метод кадрової синхронізації систем

передавання даних з нероздільним факторіальним кодуванням на основі кореляційної обробки наведеному в [27], [85]. Метод базується на властивостях відстаней Хеммінга між циклічними зсувами перестановок. В основі математичного апарату лежить вибір таких перестановок π довжини M , які задовольняють умову мінімальної відстані Хеммінга синхрокомбінації, представленої в двійковому вигляді, до всіх її циклічних зсувів повинна бути максимальною:

$$\min_{j \in [1; M-1]} d_H(\pi, \pi_j) \rightarrow \max ,$$

де d_H – відстань Хеммінга, π_j – циклічний зсув перестановки на j позицій.

Таким чином забезпечується максимальна відмінність між перестановкою та її зсувами, що дозволяє виявляти та виправляти помилки. Визначені перестановки використовуються як синхрокомбінації.

За умови довжини синхрокомбінації $M = 8$, рівномірного кодування символів перестановки з кількістю біт на символ l_r у [27], [85] визначено мінімальні відстані Хеммінга d (Таблиця 1.3).

Таблиця 1.3 – Зв'язок розподілу кількості перестановок за мінімальною відстанню Хеммінга до їх циклічних зсувів для $M=8$

Мінімальна відстань Хеммінга (d)	Кількість перестановок
0	16
4	912
6	5440
8	20752
10	13168
12	32

З визначеного розподілу кількості перестановок максимальним значенням мінімальної відстані Хеммінга для довжини перестановки $M = 8$ є

$d=12$, а кількість таких перестановок, що мають мінімальну відстань, становить 32 варіації.

З метою підвищення достовірності передавання синхрокомбінації в методі кадрової синхронізації на основі кореляційної обробки також використовується мажоритарна обробка, описана в розділі (1.3.1), накопичених бітів фрагментів прийнятих з каналу зв'язку від передавача. Відповідно кількість бітових помилок що можливо виправити складає $n \cdot \binom{l-1}{2}$ з $l \cdot n$ отриманих біт.

Після сформованої уточненої послідовності R , уточнена послідовність проходить етап кореляційної обробки. Кореляційна обробка за прийнятої перестановки π довжиною $M=8$, що має мінімальне значення відстані Хеммінга $d=12$ до циклічних зсувів обраної перестановки-синхрокомбінації π , може виправляти помилки до значення $d_{\lim} = \left\lfloor \frac{d-1}{2} \right\rfloor = 5$ включно [27], [85].

Метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням на основі кореляційної обробки викладений в роботах [27], [85] полягає в наступному.

- 1) Передавач послідовно видає синхрокомбінацію в канал зв'язку. Синхрокомбінація — це перестановка π довжиною M , де мінімальна відстань Хеммінга d до всіх її циклічних зсувів є максимальною.
- 2) Приймач накопичує K блоків, отриманих з каналу зв'язку, що складаються з l фрагментів по n біт. Значення K і l змінюються відповідно процедури, яку описано в [27], [85], [90].
- 3) Для кожного блоку уточнені послідовності R_k , де $k \in [1, K]$, розраховуються самостійно. Кожен біт цієї послідовності обчислюється за мажоритарним принципом на основі відповідних бітів отриманих фрагментів. Таким чином, якщо «одиниці» переважають у i -тих бітах фрагмента, то i -тому біту уточненої послідовності присвоюється

значення «один». Якщо ж вони містять більше «нулів», то навпаки, їм присвоюється значення «нуль».

- 4) Для кожної уточненої послідовності R_k обчислюються відстані Хеммінга для всіх циклічних зсувів синхрокомбінації. Якщо для якогось із зсувів відстань менша або дорівнює $d_{\lim} = \left\lfloor \frac{d-1}{2} \right\rfloor$, уточнена послідовність ототожнюється з циклічним зсувом, якому відповідає ця відстань.
- 5) Синхронізація встановлюється, якщо кожна з послідовностей R_k , де $k \in [1, K]$, ідентифікується одним і тим же циклічним зсувом синхрокомбінації; інакше операції з виявлення синхрокомбінації повторюються, починаючи з другого пункту в поточному списку.
- 6) Кількість накопичених фрагментів може бути збільшена послідовно до попередньо визначеного порогу $l_{\max}(1)$. Якщо синхронізація не була встановлена після досягнення цього порогу, процедура пошуку припиняється і система виводить повідомлення про збій каналу.

Імовірність встановлення синхронізації відповідно до [27], [85] визначається кількістю бітових помилок v в уточненій послідовності R , що $v \leq d_{\lim}$:

$$P_{true}(n, d_{\lim}, p_0, l) = \sum_{v=0}^{d_{\lim}} C_n^v (p_0^*)^v (1 - p_0^*)^{n-v}.$$

Імовірність встановлення хибної синхронізації у [27], [85] визначено, як теорема:

$$P_{false}(n, d_{\lim}, p_0, l) = \sum_j^{n-1} \left(\sum_{v=d_{ij}-d_{\lim}}^{d_{ij}} C_{d_{ij}}^v \left(\sum_{w=0}^{v-d_{ij}+d_{\lim}} C_{n-d_{ij}}^w (p_0^*)^{v+w} (1 - p_0^*)^{n-(v+w)} \right) \right),$$

де d_{ij} це відстань Хеммінга від синхрокомбінації π_i до її циклічного зсуву на j біт.

У роботі [27] зазначено, що у випадку процесу збільшення значення коефіцієнту накопичення фрагментів l подальшого формування уточненої

послідовності R та визначення відстаней Хеммінга на етапі кореляційної обробки. Є імовірність появи ситуації коли за невідомої ймовірності бітової помилки p_0 в каналі зв'язку величина імовірності встановлення хибної синхронізації перевищить своє допустиме значення.

Одним із шляхів зменшення ймовірності хибної синхронізації в такій ситуації є збільшення числа K блоків з l фрагментами. Зазначений підхід в роботі [27] передбачає прийом K блоків, що складаються з l фрагментів по n біт. Для кожного блоку K обчислюється незалежно уточнена послідовність R_k , $k \in [1, K]$. Встановлення синхронізму при цьому приймається, якщо всі послідовності R_k , $k \in [1, K]$ відповідають тому самому зсуву синхрокомбінації-перестановки.

Відповідно до роботи [27], ймовірність хибної синхронізації з використанням декількох незалежних блоків K для уточнених послідовностей R_k , $k \in [1, K]$ визначається як:

$$P_{false}(n, d_{lim}, p_0, L, l, K) = \sum_{j=1}^{n-1} \left(\sum_{v=d_{ij}-d_{lim}}^{d_{ij}} C_{d_{ij}}^v \sum_{w=0}^{v-d_{ij}+d_{lim}} C_{n-d_{ij}}^w (p_0^*)^{v+w} (1-p_0^*)^{n-v-w} \right)^K,$$

що має монотонний спад при збільшенні значень K і l . А ймовірність правильної синхронізації при цьому оцінюється, як:

$$P_{true}(n, d_{lim}, p_0, L, l, K) = P_{true}^K(n, d_{lim}, p_0, L) = \left(\sum_{v=0}^{d_{lim}} C_n^v (p_0^*)^v (1-p_0^*)^{n-v} \right)^K,$$

яка монотонно зростає при збільшенні значення кількості блоків K , однак зростає при збільшенні значення l .

1.3.3. Надійне передавання перестановок в каналах з високою інтенсивністю бітової помилки

Метод достовірного передавання перестановок у системах зв'язку з нероздільним факторіальним кодуванням за умови високої інтенсивності шуму в каналі передавання (близько 0.5) описаний в роботі [99] базується на кореляційній обробці перестановок викладеної в [27]. Розглянемо підхід для

забезпечення надійного передавання даних представлених у вигляді перестановок через канал зв'язку з високою інтенсивністю завад для подальшого його використання в межах одного протоколу інформаційного обміну.

Основою надійного передавання є властивість обраної перестановки у якості носія інформації – мінімальна відстань Хеммінга від циклічних зсувів перестановки носія є максимальною. Як зазначено у попередньому розділі, мінімальна відстань Хеммінга дозволяє виявляти та виправляти бітові помилки менше, або рівною значенню d_{lim} . Метод достовірного передавання перестановок також використовує мажоритарну обробку накопичених фрагментів перестановок носія. Таким чином введена надлишковість підвищує імовірність розпізнавання перестановки носія і нівелює бітові помилки, що можуть виникати в процесі передавання зашумленим каналом.

Метод достовірного передавання перестановок у системах зв'язку з нероздільним факторіальним кодуванням відповідно [99] є наступним.

1. Передавач послідовно видає в канал зв'язку перестановку W довжиною N , іменовану словом. Кожен елемент перестановки, іменований літерою L_j , $1 \leq j \leq N$, є циклічним бітовим зсувом перестановки π довжиною M , що володіє максимальним значенням мінімальної відстані Хеммінга від її n -бітного двійкового представлення до всіх її циклічних зсувів (наприклад, для $M = 8$ перестановка $\pi = (000, 001, 111, 011, 010, 101, 100, 110)$). Очевидно, що кількість циклічних зсувів перестановки π має бути не меншим за довжину перестановки W : $n \geq N$. Процедурі передавання даних передують процедура встановлення синхронізації за літерами, наприклад запропонованої в працях [27], [85].
2. Для кожної літери приймач накопичує прийняті з каналу зв'язку l фрагментів по n біт.

3. Для кожної літери незалежно обчислюється уточнена послідовностей R_j , $j \in [1, N]$. Кожен біт цієї послідовності обчислюється за мажоритарним принципом на основі відповідних біт прийнятих фрагментів.
4. Для кожної уточненої послідовності R_j обчислюються відстані Хеммінга до використовуваних передавачем літер. Якщо до якоїсь із літер ця відстань не перевищує значення $d_{\lim} = \left\lfloor \frac{d-1}{2} \right\rfloor$, j -му елементу слова встановлюється у відповідність ця літера.
5. Якщо всі послідовності R_j , $j \in [1, N]$ відповідають різним використовуваним джерелом літерам, тобто прийняте слово є перестановкою цих літер, і ця перестановка використовується джерелом, слово видають споживачеві. У іншому випадку повторюються всі операції розпізнавання слова, починаючи з п. 2 цього списку.
6. Число накопичених фрагментів може послідовно збільшуватися до деякого, заздалегідь заданого порога l_{\max} . Якщо після досягнення цього порога слово не розпізнано, процедура приймання завершується, а на вихід системи видається сигнал «Аварія каналу».

Розглянутий метод надійного передавання перестановок в каналі з високою ймовірністю бітової помилки дозволяє отримати ймовірність правильного розпізнавання перестановки що передається на рівні $P_{w_true} = 0.999$ за $p_0 = 0.495$. Однак розроблений метод не досліджувався у контексті інформаційного обміну між системами за невідомого початкового моменту передавання на етапі встановлення кадрової синхронізації. Відповідно постає актуальність у дослідженні ймовірності встановлення хибної синхронізації під дією шуму в каналі та невідомого моменту передавання синхрокомбінації. Окрему увагу становить процедура визначення межі між блоком синхрокомбінацій кадрової синхронізації та блоку даних.

1.4. Матричні протоколи шифрування та узгодження ключів

Матричні протоколи шифрування та узгодження ключів є перспективним напрямом сучасної криптографії, заснованим на використанні математичних структур матриць для перетворення інформації та обміну ключами. Перевага матричних підходів полягає у високій обчислювальній складності задачі зворотного перетворення, що ускладнює задачу дешифрування без знання ключів.

Теорія скінченних полів відіграє одну з ключових ролей у криптографії. Теоретичною базою криптографічних алгоритмів виступає алгебра скінченних полів $GF(q)$ де q – просте число або степінь простого числа. Операції над елементами цих полів (додавання, множення, інверсія) забезпечують криптографічну стійкість відповідних алгоритмів. Так, операції додавання, множення, пошуку обернених значень у симетричних алгоритмах шифрування [128], [133] реалізують над розширеним скінченним полем $GF(2^n)$. На теорії скінченних полів будують алгоритми перевірки на простоту та факторизації цілих чисел, що є фундаментом асиметричної криптографії [134], [135], [136]. Скінченні поля є невід’ємним інструментом формування електронного цифрового підпису, в тому числі на основі еліптичних кривих [137], [138].

Застосування спеціальних класів матриць при побудові криптографічних схем, таких як тропічні циркуляційні матриці або матриці Фібоначчі, дозволяють спростити процес узгодження ключів за рахунок специфічних математичних властивостей цих структур. У роботі Huang, Li та Deng [139] запропоновано новий метод криптографії з відкритим ключем, заснований на тропічних кругових матрицях (tropical circular matrices). Тропічні кругові матриці — це матриці спеціальної форми, у яких кожен рядок формується циклічним зсувом попереднього з відповідним множником t :

$$A = \begin{pmatrix} a_0 & a_{k-1} \otimes t & a_{k-2} \otimes t & \cdots & a_1 \otimes t \\ a_1 & a_0 & a_{k-1} \otimes t & \cdots & a_2 \otimes t \\ a_2 & a_1 & a_0 & \cdots & a_3 \otimes t \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{k-1} & a_{k-2} & a_{k-3} & \cdots & a_0 \end{pmatrix}.$$

Операція додавання в таких структурах виконується звичайним способом, а множення є нетривіальним перетворенням, що призводить до високої стійкості відомих криптографічних атак, включаючи атаку Котова-Ушакова (Kotov-Ushakov, KU) та RM-атаки (Rudy-Monico, RM), описаними в роботах [140] і [141] відповідно. Ключова матриця K , що виступає публічним спільним ключем формується наступним чином: $K = P_1 P_2 Y Q_1 Q_2$, де $P_1 P_2 \in C_s^k$ - тропічні матриці одного типу, $Q_1 Q_2 \in C_t^k$ - тропічні матриці іншого виду. Безпека схеми ґрунтується на складності розв'язання задачі двосторонньої дії тропічної кругової матриці (two-side tropical circular matrix action problem, TCMAP), яка є NP-важкою.

Для шифрування повідомлення M застосовується пара значень (S, R) , що формують шифртекст: $S = M + P_2 K_a Q_2$, $R = P_2 Y_2 Q_2$ де K_a - сформований ключ абонента А за формулою $K_a = P_1 Y Q_1$. Процедура дешифрування виконується як: $M = S - P_1 R Q_1$. Таким чином, правильність розшифрування гарантується властивістю комутативності дій тропічних кругових матриць.

У роботі Naseri, Abbasi та Ebrahimi Atani [10] представлено новий клас матриць M_q , які є узагальненням матриць Фібоначчі. M_q -матриці мають розмір $(q+1) \times (q+1)$ і спеціальну структуру: ненульові елементи розташовані лише на супердіагоналі та в останньому рядку. Загальний вигляд такої матриці:

$$M_q = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}.$$

Ключова властивість M_q -матриць полягає в тому, що їхній визначник обчислюється простою формулою $\det(M_q) = (-1)^q$. Ця властивість гарантує, що матриця є оборотною при всіх значеннях порядку матриці q .

У роботі [10] також запропоновано рекурсивну формулу для побудови елементів матриць, що дозволяє ефективно обчислювати степені матриці M_q^n для використання у криптографічних алгоритмах. У випадку застосування у шифруванні M_q матриця використовується як ключова матриця в модифікованій версії шифру Хілла [142]. Ключова матриця K обчислюється за формулою $K = M_\lambda^s \pmod{p}$, де s - підпис, λ - секретний ключ, p - просте число. Шифрування повідомлення відбувається за допомогою множення блоку відкритого тексту P_i на ключову матрицю K , а дешифрування операцією множення на обернену матрицю K^{-1} . Такий підхід дозволяє передавати лише два параметри (λ і s), замість повної ключової матриці, що знижує обсяг даних для передачі та підвищує безпеку за рахунок великого простору можливих ключів. Ця схема поєднує переваги шифр Хілла та процедуру обміну ключами Ель Гамала, забезпечуючи як конфіденційність, так і автентичність даних при відносній простоті реалізації.

У роботі Maxrizal [12] представлено новий підхід до побудови криптографічної системи з відкритим ключем, що базується на використанні сингулярних матриць – матриць із визначником, що дорівнює нулю. Автор пропонує замінити традиційні невироджені (несингулярні) матриці, які використовуються в багатьох існуючих схемах (наприклад, у шифрі Хілла), на сингулярні, щоб підвищити захист від атак. Головною ідеєю дослідження [12] є те, що злоумисник може знайти обернену матрицю при використанні групи невироджених матриць. Застосування сингулярної матриці, тобто такої, що не має інверсії, унеможливорює злам через пошук оберненої матриці.

Автор Maxrizal використовує систему, у якій публічна матриця X є сингулярною, а приватна матриця A залишається секретною. Формується публічна матриця шляхом множення матриць $P = XA$. На етапі формування

ключів два абоненти Аліса та Боб обирають секретні матриці A і B відповідно. Далі Аліса визначає $Y = AX$ і передає Бобу. Боб визначає $U = BX$ і також передає Алісі. Формується спільний ключ $K = UA = YB$. Для шифрування повідомлення P один з абонентів обчислює $C = K + P$, процедура дешифрування $P = C - K$.

У роботі Al-Shaarani та Gutub [19] запропоновано підхід, що поєднує матричний поділ секретів, шифрування та стеганографію для підвищення безпеки зберігання та передачі секретних даних. Схема базується на попередніх методах поділу секретів (ключів) на основі підрахунків, які генерують частки секрету шляхом заміни нулів у бітовому представленні ключа на одиниці. Автори Al-Shaarani та Gutub розширили цю ідею, використовуючи матриці для збільшення кількості можливих часток, що дозволяє забезпечити більшу гнучкість і захист. До ключових особливостей методу шифрування можна віднести: шифрування певної частки секретного ключа методом XOR; зашифровані частки приховуються в зображеннях за допомогою LSB (Least Significant Bit) або DWT (Discrete Wavelet Transform) стеганографії; таким чином, додаються два додаткові шари захисту: криптографія та стеганографія.

Сучасних досліджень та розробок нових підходів з метою забезпечення криптографічної стійкості на основі матриць є багато, зокрема можна ще виділити працю [143], що вивчає технологію хаотичного шифрування зображення та застосування теорії напівтензорного добутку матриць; авторів дослідження [6], де запропонована техніка гомоморфного шифрування на основі перетворень матриці зі зсувами, обертаннями та транспозиціями; робота [7] демонструє створення цифрового підпису на основі криптосистеми McEliece з використанням випадкових зворотних матриць.

Окремо можна виділити трьохетапний механізм випадкового матричного шифрування Шаміра від автора Francois Dupont [144], що використовує трьохетапний протокол з операторами шифрування, які є випадковими матрицями. У роботі запропоновано новий варіант механізму

трьохпрохідного шифрування Шаміра, що використовує випадкові комутативні матриці для захищеного обміну повідомленнями без попереднього обміну ключами.

Метод на основі трьохпрохідного випадкового матричного шифрування Шаміра [144] дозволяє двом учасникам (Алісі та Бобу) зашифрувати повідомлення через послідовне множення на їхні власні випадкові матриці, які мають властивість комутативності (тобто $AB = BA$). Математична основа методу ґрунтується на представленні комутативної матриці G через спектральний розклад $G = C \cdot D \cdot C^{-1}$, де всі комутативні матриці мають спільну матрицю власних векторів C , D - діагональна матриця власних значень. Ідея використання власних векторів та власних значень забезпечують комутативність сформованих матриць. Проте, в цій праці автор обмежується операціями над комутативними матрицями, використовуючи обернені, циркуляційні, перестановочні матриці, однак не розширює свої дослідження на поля матриць.

Скінченні поля матриць мають потенційну перспективу використання в криптографічних схемах перетворення інформації, обміну ключами та цифрового підпису. І можуть бути потенційно ефективним використанням скінченних полів матриць у системах передавання даних на основі перестановок [17], [27], [145]. Однак довжини ключів і інформаційних блоків для протоколів на основі матричних перетворень, і для протоколів факторіального кодування в загальному випадку не співпадають, відображення матриць у перестановки не є бієктивним (сюр'єктивним і ін'єктивним одночасно) відповідно до визначення наведеного в [146]. Відповідно, виникає потреба в формуванні та дослідженні алгоритмів відображення матриць у перестановки. Розглянуті роботи з використанням матричної алгебри демонструють значні перспективи в розробці нових криптографічних алгоритмів, що відповідають сучасним вимогам безпеки інформаційного обміну, та відкривають широкі можливості для подальших теоретичних та експериментальних досліджень у цьому напрямі.

1.5. Цілі та задачі дисертаційного дослідження

Метою цього дисертаційного дослідження є забезпечення захищеного інформаційного обміну в системах з нероздільним факторіальним кодуванням даних в умовах зашумлених каналів зв'язку.

Використання нероздільних факторіальних кодів має переваги та особливості при побудові стійких систем інформаційного обміну. Актуальною є задача підвищення достовірності синхронізації при використанні існуючих методів нероздільного факторіального кодування, а також розширення існуючих методів кадрової синхронізації, що дозволить використовувати факторіальні коди у каналах з високою імовірністю бітової помилки.

Під час розгляду методів кадрової синхронізації [16], [27], [85], [90] зазначається, що в передавач надходять фрагменти-синхрокомбінації уражені дією завади в каналі передавання і зсунуті за фазою у регістрі для їх накопичення. При цьому, не враховується невизначений момент початку передавання синхрокомбінації від передавача або можливість повної відсутності передавання через канал зв'язку. Синхронізація здійснюється шляхом поетапного накопичення фрагментів; у разі досягнення заданого порогового значення їх кількості $l_{\max}(l)$ та за відсутності підтвердженої синхронізації, процедура синхронізації виконується в межах встановленого часового інтервалу.

За вищезазначеними умовами у разі відсутності передавання синхрокомбінації каналом зв'язку, біти шуму можуть сприйматися приймачем як коректна інформація. Реалізація процесу встановлення синхронізму за розглянутими методами [27], [85], [90] може призвести до хибних спрацювань підсистеми синхронізації або спричинити перехід приймальної сторони в аварійний режим (так звану «аварію каналу») за фактичної відсутності передавання. Тому практичне та коректне застосування розглянутих методів кадрової синхронізації в системах з нероздільним факторіальним кодуванням

— зокрема, методів на основі кореляційної обробки чи розділення кодового слова на префіксну та суфіксну частини — є можливим лише за наявності зворотного каналу зв'язку між передавачем і приймачем. Однак у роботах [16], [27], [85], [90] деталі організації такого зворотного каналу не розкриваються.

Обмеження методів кадрової синхронізації полягають у тому, що не можуть ефективно працювати за невідомого моменту початку приймання синхрокомбінації у канал зв'язку з високим значенням імовірності появи бітової помилки. Використання адаптивного алгоритму накопичення фрагментів з каналу передавання дозволяє враховувати якість каналу і тим самим зменшувати накладні витрати за часом на етапі синхронізації. Процедура реакції приймача у системі інформаційного обміну на етапі встановлення синхронізації за невідомого початкового моменту приймання має невизначеність через вплив шуму в каналі і може призводити до некоректного відпрацювання встановлення синхронізму.

Використання матричних структур у сучасній криптографії демонструють значний потенціал завдяки своїм алгебраїчним властивостям і високій обчислювальній складності зворотних операцій. Перспектива застосування скінченних полів у поєднанні з матричними структурами, відкриває нові можливості для побудови стійких криптографічних схем та ефективного узгодження ключів. Особливо актуальним є використання таких структур у системах з нероздільним факторіальним кодуванням, де коди базуються на перестановках і потребують спеціальних механізмів узгодження з метою забезпечення захищеного інформаційного обміну.

Забезпечення надійної синхронізації є критично важливим для стабільної роботи систем інформаційного обміну, особливо в умовах зашумлених каналів та невизначеного моменту передавання. Існуючі методи кадрової синхронізації в цілому демонструють позитивні результати, проте потребують подальшої перевірки та удосконалення для підвищення стійкості до бітових помилок і запобігання хибним спрацюванням. У цьому контексті особливої уваги заслуговує дослідження ефективності удосконалених підходів

до синхронізації шляхом моделювання та експериментальної перевірки їхньої роботи в умовах, наближених до реальних.

Виходячи з цього, задачами роботи є:

- розвинути метод кадрової синхронізації нероздільних факторіальних кодів для випадків за невідомого моменту початку приймання синхрокомбінацій передавача;
- розвинути математичну модель процесу виявлення синхрокомбінації систем передавання інформації з нероздільним факторіальним кодуванням даних за невідомого моменту початку приймання синхрокомбінацій передавача;
- розробити математичну модель скінченного поля квадратних матриць порядку 2 над скінченим полем простих чисел та розробити методику їх використання для узгодження ключів-перестановок;
- побудувати імітаційну модель системи інформаційного обміну, провести порівняльні експериментальні оцінки удосконаленого методу кадрової синхронізації.

Поставлені задачі дисертаційного дослідження несуть наукову новизну і практичну цінність, а їх вирішення дозволить досягти поставленої мети.

1.6. Висновки

У першому розділі дисертаційної роботи здійснено аналіз сучасного стану захищеного інформаційного обміну в умовах зашумлених каналів зв'язку з використанням НФКД. Встановлено, що класичні підходи до захисту інформації мають обмеження при використанні в умовах високої інтенсивності бітових помилок та невизначеного моменту початку приймання синхрокомбінації.

Розглянуто сучасні досягнення у сфері матричної криптографії та нероздільного факторіального кодування даних, які поєднують властивості самосинхронізації, стійкість до помилок і високу обчислювальну складність

зворотного перетворення. Встановлено перспективність використання квадратних матриць у криптографічних перетвореннях та необхідності синтезу математичної моделі квадратних матриць над скінченним полем простих чисел з метою їх використання при реалізації узгодження ключів криптографічних протоколах.

Проведено аналіз існуючих методів кадрової синхронізації на основі перестановок. Показано, що існуючі підходи потребують удосконалення для забезпечення надійної синхронізації в умовах невизначеного початку передавання та високого рівня шуму у каналі зв'язку.

Встановлені цілі і сучасний стан області дослідження формують підґрунтя для напряму подальших досліджень, що направлені на синтезі нових математичних моделей, методів синхронізації та використання імітаційних моделей з метою перевірки та вдосконалення ефективності нових підходів.

2. МЕТОД КАДРОВОЇ СИНХРОНІЗАЦІЇ СИСТЕМ ПЕРЕДАВАННЯ ДАНИХ З НЕРОЗДІЛЬНИМ ФАКТОРІАЛЬНИМ КОДУВАННЯМ

2.1. Вступ

У другому розділі розглядається метод кадрової синхронізації в системах передавання даних із нероздільним факторіальним кодуванням. Основна увага приділена вирішенню завдання синхронізації при умові високого рівня шуму в каналі зв'язку, коли момент початку прийому даних є невідомим. Технічний результат досягнуто за рахунок використання ковзного вікна.

Для забезпечення необхідної ймовірності правильної та хибної синхронізації запропоновано підхід, що враховує результати роботи підсистеми синхронізації для послідовних зсувів бітів в ковзному вікні. Розглянуто механізм застосування ковзного вікна фіксованого розміру, що дозволяє здійснювати безперервний аналіз вхідного потоку даних.

У межах розділу:

- розроблено метод кадрової синхронізації на основі ковзного вікна з урахуванням результатів роботи підсистеми синхронізації для послідовних зсувів цього ковзного вікна;
- оцінено ймовірнісні характеристики правильної та хибної кадрової синхронізації;
- проаналізовано вплив параметрів синхронізації на показник імовірності хибної кадрової синхронізації;
- розроблено алгоритми та імітаційну модель підсистеми синхронізації, що підтверджують ефективність запропонованого методу в умовах високого рівня шуму.

Отримані результати можуть бути використані для підвищення завадостійкості інформаційного обміну в каналах зв'язку з високою ймовірністю бітової помилки, а також у криптографічних протоколах, що використовують перестановки для передавання даних.

2.2. Схема кадрової синхронізації з використанням ковзного вікна

Перший етап інформаційного обміну між передавачем та отримувачем полягає у встановленні кадрової синхронізації перестановок, що передаються. Розроблений у дисертаційному дослідженні метод кадрової синхронізації базується на підході, запропонованому в працях [147], [27], які використовують як синхрокомбінацію перестановку, що має максимальне значення мінімальної відстані Хеммінга від її двійкового представлення до всіх циклічних зсувів такого представлення. Такий підхід передбачає, що отримувач накопичує з каналу зв'язку K блоків по l фрагментів із M елементів з наступною мажоритарною [148], [149] і кореляційною обробкою [150], [151], [152] накопичених фрагментів. Значення K і l змінюються відповідно до визначеної в [27] методики. Попередньо встановлений мінімальний поріг ймовірності правильної синхронізації P_{true} визначає достатню кількість накопичених фрагментів.

Метод кадрової синхронізації [147], [27] передбачає послідовне передавання в канал зв'язку синхрокомбінації. Наприклад, для $M = 8$ таким синхрокомбінацією є перестановка $\pi = (000, 001, 111, 011, 010, 101, 100, 110)$ з точністю до її циклічного зсуву на число біт, кратне $l_r = 3$, інверсії біт, та зворотного порядку їх слідування.

Висока інтенсивність шуму в каналі передавання призводить до того, що приймач неспроможний визначити початковий момент передавання синхрокомбінації передавачем. Для забезпечення надійного передавання при інтенсивному шумі алгоритм виявлення меж синхрокомбінацій дещо змінюється.

Відповідно до [27] достатня кількість накопичених фрагментів для забезпечення мінімального значення ймовірності правильної синхронізації P_{true_min} вибирається як мінімальне значення l , за якого ймовірність правильної синхронізації для $K = 1$ не менша заданого P_{true_min} . У роботі [27] це значення

кількості накопичених фрагментів позначено як $l_{\max}(1)$. В подальшому описі цього розділу кількість накопичених фрагментів позначатиметься через l_{\max} .

Виходячи із сказаного вище, а також того, що початковий момент передавання синхрокомбінації невідомий, приймачеві для пошуку синхронізму доцільно використовувати ковзне вікно із шириною l_{\max} фрагментів (Рисунок 2.1).

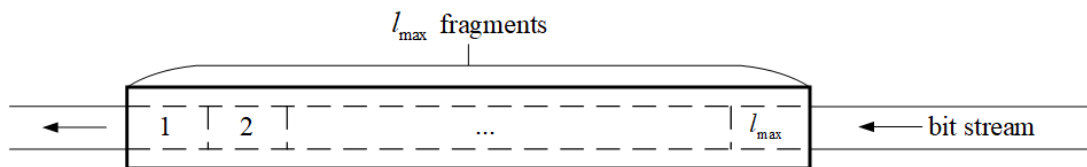


Рисунок 2.1 – Схема використання ковзного вікна з l_{\max} фрагментів

Таким чином, приймач, має потік бітів в ковзному вікні, який зміщується на один біт, постійно аналізує отримані l_{\max} фрагментів з каналу зв'язку і намагається встановити кадрову синхронізацію. У цьому випадку динамічна зміна значень K і l , запропонована в [27], не має сенсу. Відповідно, математична модель процесу приймання синхрокомбінацій відрізняється від викладеної в [27].

Приймачеві доводиться постійно слухати канал. За відсутності сигналу від передавача в ковзному вікні присутній тільки шум. Відповідно, ймовірність бітової помилки становить 0.5.

Після того, як передавач починає видавати в канал зв'язку службові сигнали для тактової та кадрової синхронізації, у ковзному вікні системи синхронізації приймача починають з'являтися фрагменти із синхрокомбінаціями (Рисунок 2.2)

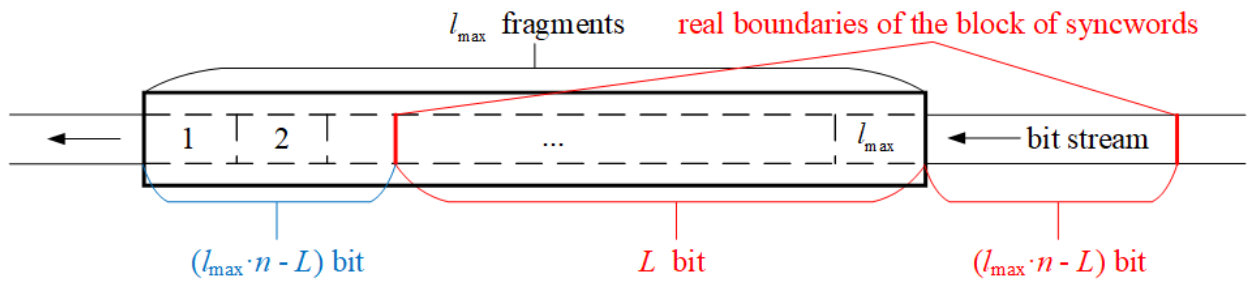


Рисунок 2.2 – Схема заповнення ковзного вікна даними від передавача

На схемі (Рисунок 2.2) зображено ковзне вікно у якому знаходяться $l_{\max} \cdot n$ біт синхрокомбінацій передавача. Тоді для більшої наочності процесу мажоритарного прийому накопичених біт представимо фрагменти в ковзного вікна у вигляді, зображеному на (Рисунок 2.3). При цьому заштриховані області містять тільки біти шуму (імовірність помилки – 0.5), незаштриховані – містять біти синхрокомбінацій передавача (імовірність помилки – p_0).

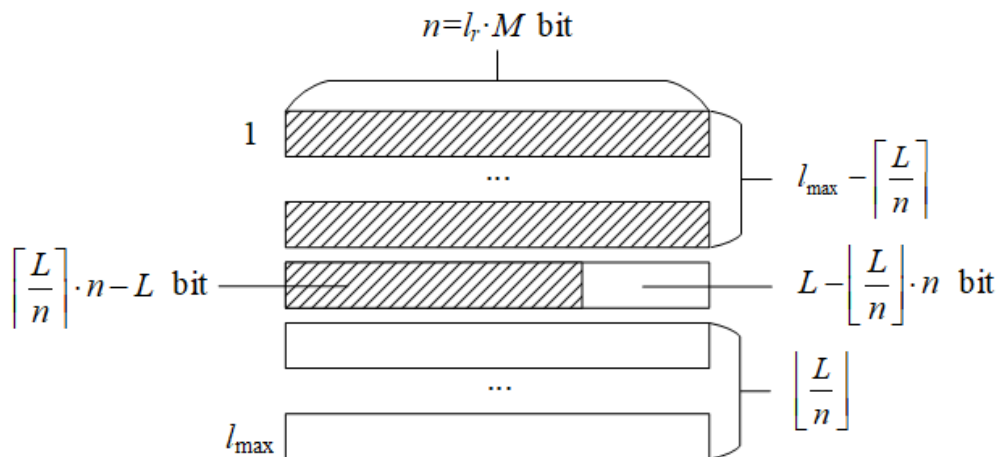


Рисунок 2.3 – Схема мажоритарної обробки накопичених біт

За накопиченими фрагментами мажоритарно обчислюється уточнена послідовність R , де частина помилок (якщо вони є) виправлена.

2.3. Оцінки ймовірнісних показників кадрової синхронізації

Імовірності правильної та хибної синхронізації залежать від імовірності бітової помилки p_0 після мажоритарної обробки l_{\max} прийнятих фрагментів.

Виходячи з того, що приймач не знає початкового моменту передавання синхрокомбінації, це призводить до наступного.

Кількість біт синхрокомбінації від передавача, що знаходяться в ковзному вікні системи синхронізації приймача, може бути не кратна довжині кодової комбінації $l_r = \lceil \log_2 M \rceil$, як показано на (Рисунок 2.3). Тоді ймовірність бітової помилки в уточненій послідовності після мажоритарної обробки прийнятих l_{\max} фрагментів можна оцінити таким чином:

$$p_0^* \leq \sum_{i=0}^{l_1} \left(C_{l_1}^i p_0^i (1-p_0)^{l_1-i} \cdot \sum_{j=(l_{\max}+1)/2-i}^{l_{\max}-l_1} C_{l_{\max}-l_1}^j (0.5)^{l_{\max}-l_1} \right) \quad (2.1)$$

$$\text{для } L < n \cdot \frac{l_{\max} + 3}{2} \text{ і}$$

$$p_0^* \leq \sum_{i=0}^{l_{\max}-l_1} \left(C_{l_{\max}-l_1}^i (0.5)^{l_{\max}-l_1} \cdot \sum_{j=(l_{\max}+1)/2-i}^{l_1} C_{l_1}^j p_0^j (1-p_0)^{l_1-j} \right) \quad (2.2)$$

$$\text{для } L \geq n \cdot \frac{l_{\max} + 3}{2},$$

$$\text{де } l_1 = \left\lfloor \frac{L}{n} \right\rfloor - \text{кількість цілих фрагментів, що містять тільки біти}$$

синхрокомбінації від передавача (які ймовірно, уражені помилкою).

Оцінки (2.1) і (2.2) сформовано шляхом заміни фрагмента, в якому присутні біти шуму і біти синхрокомбінації, фрагментом лише з бітами шуму, а також з урахуванням того, що $p_0 < 0.5$.

У праці [92] визначено, що для $M = 8$ і $p_0 = 0.4$ значення $l_{\max} = 75$. Для цих параметрів $M = 8$ і $p_0 = 0.4$ сформовано графік залежності (Рисунок 2.4) оцінки ймовірності бітової помилки в уточненій послідовності R від значення $L \in [0; 75 \cdot 24]$.

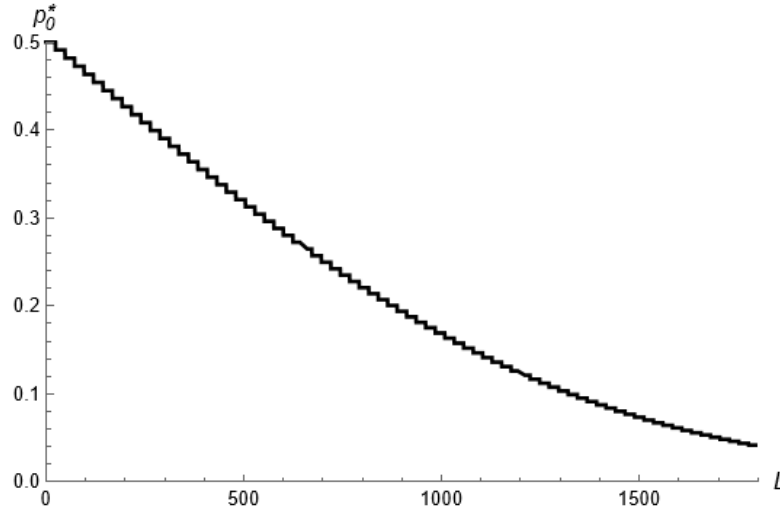


Рисунок 2.4 – Графік залежності оцінки ймовірності бітової помилки в уточненій послідовності від числа біт синхрокомбінацій у ковзному вікні при $M = 8$ і $p_0 = 0.4$

Графік має ступінчастий характер через спрощене обчислення оцінки зверху (2.1) і (2.2). Відповідно до графіку для $L = 0$ значення $p_0^* = 0.5$, а для $L = 1800$ – $p_0^* = 0.0396$. Для обчислення точного значення ймовірності бітової помилки в уточненій послідовності використано наступне твердження.

Твердження 1. Імовірність бітової помилки в уточненій послідовності R після приймання L біт синхрокомбінації передавача дорівнює:

1) для $L \leq n \cdot \frac{l_{\max} + 1}{2}$:

$$p_0^* = \left(1 + l_1 - \frac{L}{n}\right) \cdot \sum_{i=0}^{l_1} \left(C_{l_1}^i p_0^i (1 - p_0)^{l_1-i} \cdot \sum_{j=(l_{\max}+1)/2-i}^{l_{\max}-l_1} C_{l_{\max}-l_1}^j (0.5)^{l_{\max}-l_1} \right) + \left(\frac{L}{n} - l_1 \right) \cdot \sum_{i=0}^{l_1+1} \left(C_{l_1+1}^i p_0^i (1 - p_0)^{l_1+1-i} \cdot \sum_{j=(l_{\max}+1)/2-i}^{l_{\max}-l_1-1} C_{l_{\max}-l_1-1}^j (0.5)^{l_{\max}-l_1-1} \right); \quad (2.3)$$

2) для $L > n \cdot \frac{l_{\max} + 1}{2}$:

$$p_0^* = \left(\frac{L}{n} - l_1 \right) \sum_{i=0}^{l_{\max}-l_1-1} \left(C_{l_{\max}-l_1-1}^i (0.5)^{l_{\max}-l_1-1} \cdot \sum_{j=(l_{\max}+1)/2-i}^{l_1+1} C_{l_1+1}^j p_0^j (1 - p_0)^{l_1+1-j} \right) + \left(1 + l_1 - \frac{L}{n} \right) \cdot \sum_{i=0}^{l_{\max}-l_1} \left(C_{l_{\max}-l_1}^i (0.5)^{l_{\max}-l_1} \cdot \sum_{j=(l_{\max}+1)/2-i}^{l_1} C_{l_1}^j p_0^j (1 - p_0)^{l_1-j} \right). \quad (2.4)$$

Доведення.

За даними, наведеними на (Рисунок 2.3), кількість біт синхрокомбінації передавача L у кожному вікні в загальному випадку не кратна довжині фрагмента $l_r \cdot M$, у $L - \left\lfloor \frac{L}{n} \right\rfloor \cdot n$ випадках з n для формування біта за мажоритарним принципом беруть участь $\left\lfloor \frac{L}{n} \right\rfloor + 1$ біт синхрокомбінації передавача і $l_{\max} - \left\lfloor \frac{L}{n} \right\rfloor - 1$ біт шуму. Відповідно, у $n - L + \left\lfloor \frac{L}{n} \right\rfloor \cdot n$ випадках з n для формування біта за мажоритарним принципом беруть участь $\left\lfloor \frac{L}{n} \right\rfloor$ біт синхрокомбінації передавача і $l_{\max} - \left\lfloor \frac{L}{n} \right\rfloor$ біт шуму. З урахуванням того, що бітова помилка виникає тоді, коли кількість помилок у відповідних бітах накопичених фрагментів не менша за значення $\frac{l_{\max} + 1}{2}$, впливають необхідні вирази ймовірності бітової помилки (2.3) і (2.4) для $L \leq n \cdot \frac{l_{\max} + 1}{2}$ і $L > n \cdot \frac{l_{\max} + 1}{2}$ відповідно.

Графік залежності ймовірності бітової помилки в уточненій послідовності R від значення $L \in [0, 75 \cdot 24]$ при $M = 8$ і $p_0 = 0.4$ наведено на рисунку 2.5.

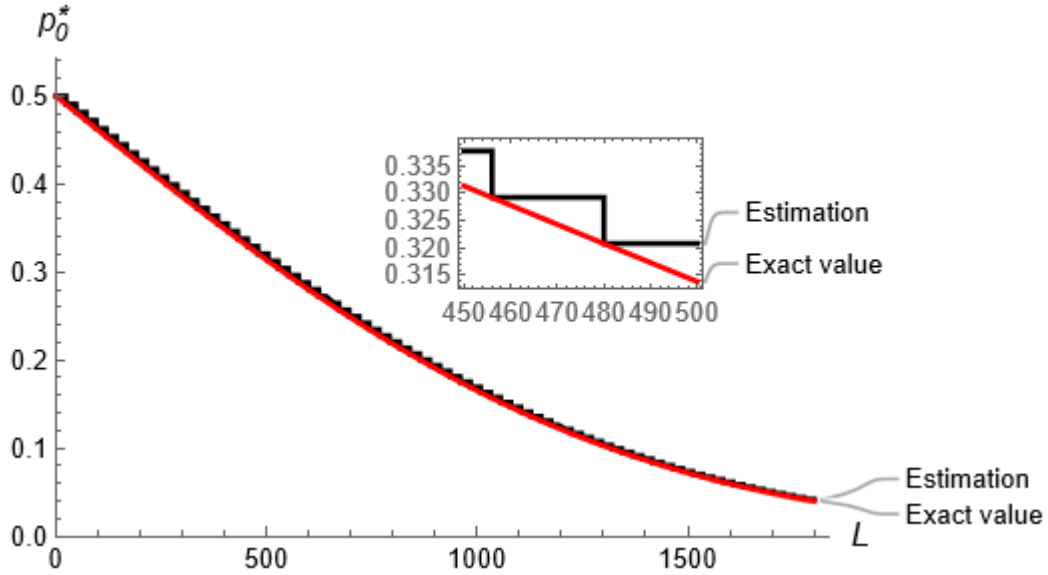


Рисунок 2.5 – Графік залежності ймовірності бітової помилки в уточненій послідовності від числа біт синхрокомбінації у кожному вікні при $M = 8$ і $p_0 = 0.4$

Для оцінки ймовірностей правильної та хибної синхронізації використано вирази, визначені в роботі [27]:

$$P_{true}(n, d_{lim}, p_0, L) \approx \sum_{v=0}^{d_{lim}} C_n^v (p_0^*)^v (1 - p_0^*)^{n-v}, \quad (2.5)$$

$$P_{false}(n, d_{lim}, p_0, L) \approx \sum_{j=1}^{n-1} \left(\sum_{v=d_{ij}-d_{lim}}^{d_{ij}} C_{d_{ij}}^v \left(\sum_{w=0}^{v-d_{ij}+d_{lim}} C_{n-d_{ij}}^w (p_0^*)^{v+w} (1 - p_0^*)^{n-(v+w)} \right) \right). \quad (2.6)$$

Для синхрокомбінації $\pi = (000, 001, 111, 011, 010, 101, 100, 110)$ вираз (2.5)

має вигляд: $P_{true}(24, 5, p_0, L) \approx \sum_{v=0}^5 C_{24}^v (p_0^*)^v (1 - p_0^*)^{24-v}$, а (2.6)

$$P_{false}(24, 5, p_0, L) \approx 19 \sum_{v=7}^{12} C_{12}^v \left(\sum_{w=0}^{v-7} C_{12}^w (p_0^*)^{v+w} (1 - p_0^*)^{24-v-w} \right) + 2 \sum_{v=9}^{14} C_{14}^v \left(\sum_{w=0}^{v-9} C_{10}^w (p_0^*)^{v+w} (1 - p_0^*)^{24-v-w} \right) + 2 \sum_{v=11}^{16} C_{16}^v \left(\sum_{w=0}^{v-11} C_8^w (p_0^*)^{v+w} (1 - p_0^*)^{24-v-w} \right).$$

Графіки функцій виразів (2.5) і (2.6) в залежності від L при $M = 8$ і $p_0 = 0.4$ наведено на (Рисунок 2.6, Рисунок 2.7).

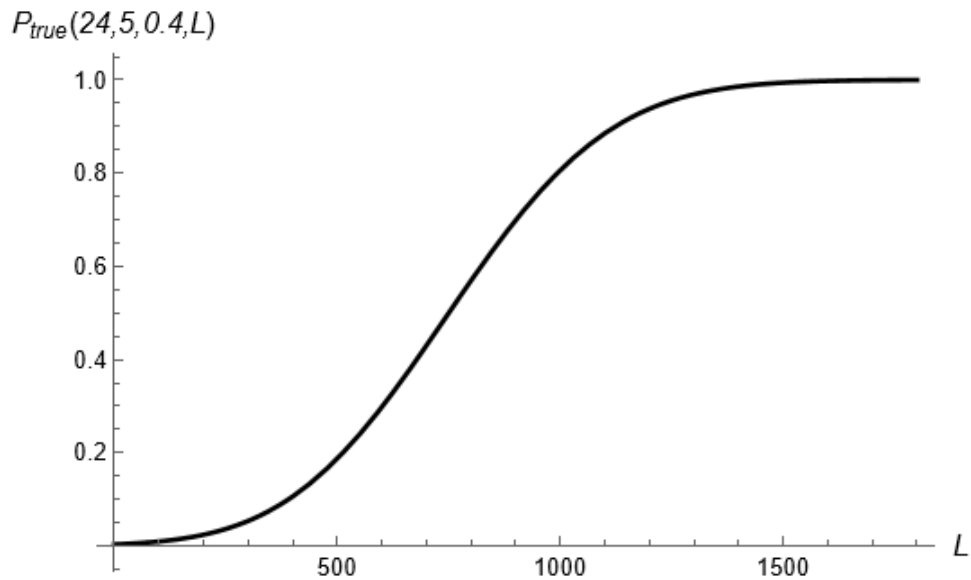


Рисунок 2.6 – Графік залежності оцінки ймовірності правильної синхронізації від числа біт синхрокомбінації у ковзному вікні для $M = 8$ і $p_0 = 0.4$

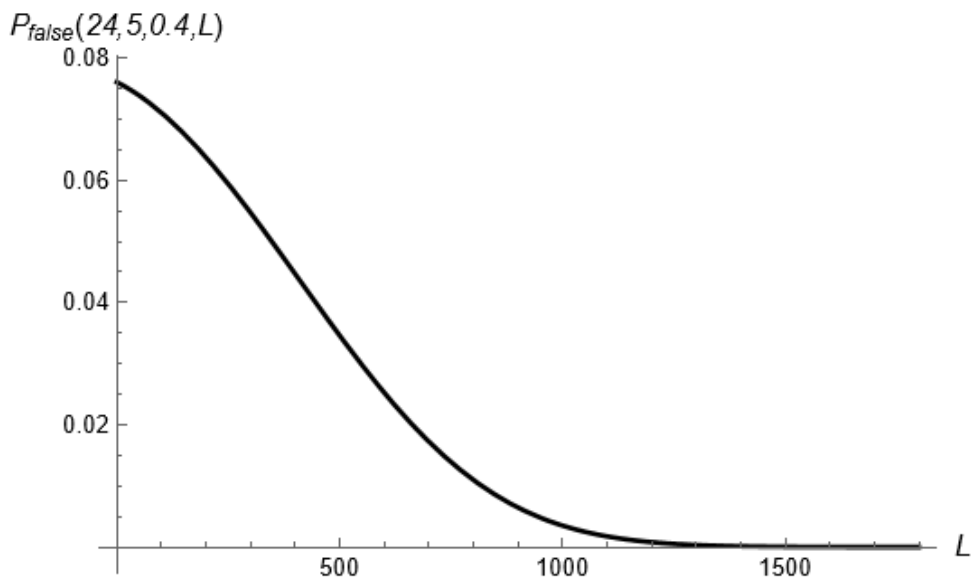


Рисунок 2.7 – Графік залежності оцінки ймовірності хибної синхронізації від числа біт синхрокомбінації у ковзному вікні для $M = 8$ і $p_0 = 0.4$

Графіки (Рисунок 2.6, Рисунок 2.7) показують, що ймовірність правильної синхронізації збільшується від 0.0033 при $L=0$ до 0.9997 для $L=1800$, в той час, як ймовірність помилкової синхронізації зменшується з 0.076 при $L=0$ до $1.198 \cdot 10^{-6}$ для $L=1800$.

З метою експериментального підтвердження сформованих залежностей $P_{true}(24,5,0.4,L)$ та $P_{false}(24,5,0.4,L)$ проведено комп'ютерне імітаційне моделювання процесу роботи системи кадрової синхронізації у результаті визначено відносні частоти $W_{true}(24,5,0.4,L)$ правильної та $W_{false}(24,5,0.4,L)$ хибної синхронізації. Для кожного значення L виконано 1000 випробувань. Графіки досліджених залежностей представлені на рисунках (Рисунок 2.8, Рисунок 2.9).

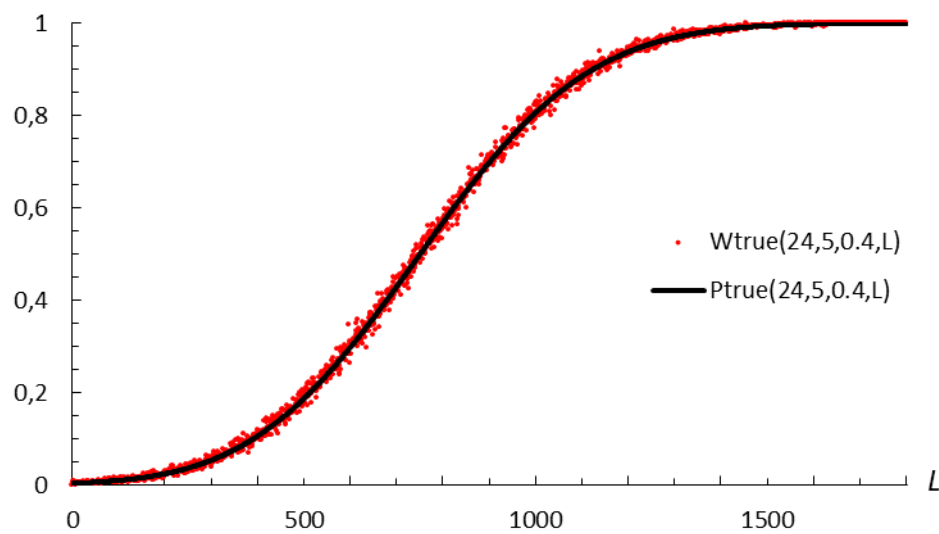


Рисунок 2.8 – Графік залежності $P_{true}(24,5,0.4,L)$ та $W_{true}(24,5,0.4,L)$

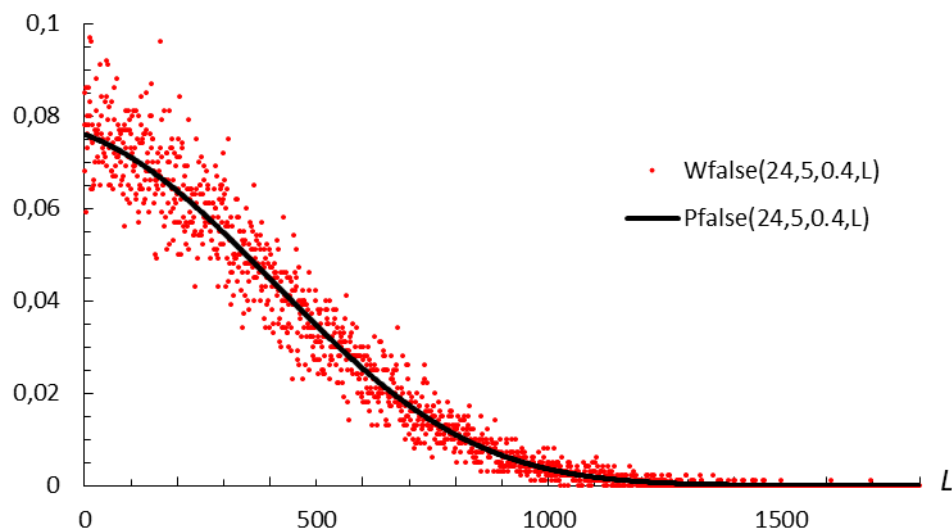


Рисунок 2.9 – Графік залежності $P_{false}(24,5,0.4,L)$ та $W_{false}(24,5,0.4,L)$

Порівняльна візуальна оцінка $P_{true}(24,5,0.4,L)$ і $W_{true}(24,5,0.4,L)$, а також $P_{false}(24,5,0.4,L)$ і $W_{false}(24,5,0.4,L)$ свідчить про коректність виразів (2.5) і (2.6), а також про адекватність розробленої моделі.

За аналогією з [27]:

- імовірність правильної синхронізації P_{true_final} після прийому L біт синхрокомбінації може бути оцінена знизу:

$$P_{true_final}(n, d_{lim}, p_0, L) \geq P_{true}(n, d_{lim}, p_0, L); \quad (2.7)$$

- імовірність хибної синхронізації P_{false_final} після приймання L біт синхрокомбінації може бути оцінена зверху сумою ймовірностей хибної синхронізації для всіх значень кількості накопичених біт синхрокомбінації, менших L :

$$P_{false_final}(n, d_{lim}, p_0, L) \leq \sum_{j=0}^L P_{false}(n, d_{lim}, p_0, j). \quad (2.8)$$

Таким чином, імовірність хибної синхронізації є високою. Рисунок 2.10 демонструє результати експериментального дослідження відносних частот правильної W_{true} та хибної W_{false} синхронізації для $M = 8$ і $p_0 = 0.4$.

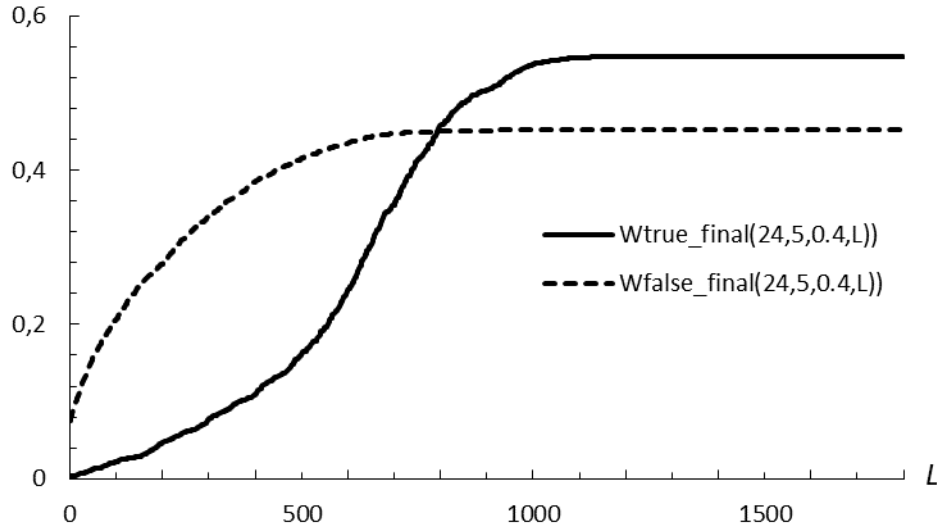


Рисунок 2.10 – Графік залежності відносних частот правильної та хибної синхронізації від числа біт синхрокомбінації у ковзному вікні для $M = 8$ і $p_0 = 0.4$

$$p_0 = 0.4$$

2.4. Підхід до зменшення ймовірності хибної синхронізації

Одним із шляхів зменшення ймовірності хибної синхронізації є збільшення числа K блоків за l фрагментами. Такий підхід [27] до збільшення значення K передбачає прийом K блоків, що складаються з l фрагментів по n біт. Для кожного блоку незалежно обчислюють уточнену послідовність R_k , $k \in [1, K]$. Якщо всі послідовності R_k , $k \in [1, K]$, асоціюються з одним і тим самим зсувом синхрокомбінації, відповідний блок системи циклової синхронізації приймає рішення про встановлений синхронізм.

Відповідно до [27], [92], ймовірність помилкової синхронізації для K блоків по l фрагментів становить:

$$P_{false}(n, d_{lim}, p_0, L, l, K) = \sum_{j=1}^{n-1} \left(\sum_{v=d_{ij}-d_{lim}}^{d_{ij}} C_{d_{ij}}^v \sum_{w=0}^{v-d_{ij}+d_{lim}} C_{n-d_{ij}}^w (p_0^*)^{v+w} (1-p_0^*)^{n-v-w} \right)^K, \quad (2.9)$$

яка монотонно зменшується зі збільшенням як значення K , так і значення l .

З іншого боку, можливість правильної синхронізації:

$$P_{true}(n, d_{lim}, p_0, L, l, K) = P_{true}^K(n, d_{lim}, p_0, L) = \left(\sum_{v=0}^{d_{lim}} C_n^v (p_0^*)^v (1 - p_0^*)^{n-v} \right)^K \quad (2.10)$$

монотонно зменшується зі збільшенням K , але збільшується зі збільшенням l . Водночас, $L \in [0; l_r \cdot M \cdot l \cdot K]$, а ймовірність бітової помилки в уточнених послідовностях R_k після приймання L біт синхрокомбінації передавача може бути оцінена шляхом модифікації виразів (2.3) і (2.4) наступним чином:

$$1) \text{ для } \left\lfloor \frac{L}{K} \right\rfloor \leq n \cdot \frac{l+1}{2}:$$

$$p_0^* \leq \left(1 + l_1 - \left\lfloor \frac{L}{K} \right\rfloor \cdot \frac{1}{n} \right) \cdot \sum_{i=0}^{l_1} \left(C_{l_1}^i p_0^i (1 - p_0)^{l_1-i} \cdot \sum_{j=(l+1)/2-i}^{l-l_1} C_{l-l_1}^j (0.5)^{l-l_1} \right) + \left(\left\lfloor \frac{L}{K} \right\rfloor \cdot \frac{1}{n} - l_1 \right) \cdot \sum_{i=0}^{l_1+1} \left(C_{l_1+1}^i p_0^i (1 - p_0)^{l_1+1-i} \cdot \sum_{j=(l+1)/2-i}^{l-l_1-1} C_{l-l_1-1}^j (0.5)^{l-l_1-1} \right); \quad (2.11)$$

$$2) \text{ для } \left\lfloor \frac{L}{K} \right\rfloor > n \cdot \frac{l+1}{2}:$$

$$p_0^* \leq \left(\left\lfloor \frac{L}{K} \right\rfloor \cdot \frac{1}{n} - l_1 \right) \sum_{i=0}^{l-l_1-1} \left(C_{l-l_1-1}^i (0.5)^{l-l_1-1} \cdot \sum_{j=(l+1)/2-i}^{l_1+1} C_{l_1+1}^j p_0^j (1 - p_0)^{l_1+1-j} \right) + \left(1 + l_1 - \left\lfloor \frac{L}{K} \right\rfloor \cdot \frac{1}{n} \right) \cdot \sum_{i=0}^{l-l_1} \left(C_{l-l_1}^i (0.5)^{l-l_1} \cdot \sum_{j=(l+1)/2-i}^{l_1} C_{l_1}^j p_0^j (1 - p_0)^{l_1-j} \right); \quad (2.12)$$

$$\text{де } l_1 = \left\lfloor \frac{L}{K \cdot n} \right\rfloor.$$

Для того, щоб імовірність хибної синхронізації не перевищувала свого граничного значення P_{false_max} , імовірність правильної синхронізації була не менше свого граничного значення P_{true_min} , а також щоб правильна синхронізація відбувалася якнайшвидше, необхідно визначити пару значень K і l , за яких $P_{false_final}(n, d_{lim}, p_0, L, l, K) \leq P_{false_max}$, $P_{true_final}(n, d_{lim}, p_0, L, l, K) \geq P_{true_min}$ а $K \cdot l$ набуває мінімального значення. Разом з тим, $P_{false_final}(n, d_{lim}, p_0, L, l, K)$ та $P_{true_final}(n, d_{lim}, p_0, L, l, K)$ оцінюють так:

$$P_{false_final}(n, d_{lim}, p_0, L, l, K) \leq \sum_{j=0}^L P_{false}(n, d_{lim}, p_0, j, l, K); \quad (2.13)$$

$$P_{true_final}(n, d_{lim}, p_0, L, l, K) \geq P_{true}(n, d_{lim}, p_0, L, l, K). \quad (2.14)$$

Значення $l_{true_min}(K)$ буде мінімальною кількістю фрагментів у кожному з K блоків, за якого $P_{true_final}(n, d_{lim}, p_0, L_{max}(K), l_{true_min}(K), K) \geq P_{true_min}$, $L_{max}(K) = l_r \cdot M \cdot l_{true_min}(K) \cdot K$. Тоді завдання визначення пари $(K; l)$ значень полягає в наступному:

- 1) задають значення n, d_{lim}, p_0 , початкове значення $K = 1$;
- 2) обчислюють значення $l_{true_min}(K)$, для якого $P_{true_final}(n, d_{lim}, p_0, L_{max}(K), l_{true_min}(K), K) \geq P_{true_min}$, де $L_{max}(K) = l_r \cdot M \cdot l_{true_min}(K) \cdot K$;
- 3) якщо $P_{false_final}(n, d_{lim}, p_0, L_{max}(K), l_{true_min}(K), K) \leq P_{false_max}$, пара значень $(K; l_{true_min}(K))$ може бути використана в якості необхідних параметрів процедури синхронізації;
- 4) якщо ж $P_{false_final}(n, d_{lim}, p_0, L_{max}(K), l_{true_min}(K), K) > P_{false_max}$, виконується перехід до пункту 2, попередньо збільшивши значення K на одиницю.

Обчислені і зведені значення $l_{true_min}(K)$ для $K = 1, 2, 3, \dots$ за значень $P_{true_min} = 0.9997$ і $P_{false_max} = 0.0003$ наведено в (Таблиця 2.1).

Таблиця 2.1 – Характеристики підходу до зменшення ймовірності хибної синхронізації

K	1	2	3	4
$l_{true_min}(K)$	75	81	83	85
$P_{false_final}(n, d_{lim}, p_0, L_{max}(K), l_{true_min}(K), K)$	1	0.182	$7.4 \cdot 10^{-4}$	$2.9 \cdot 10^{-6}$

Графіки залежності $P_{false_final}(n, d_{lim}, p_0, L, l_{true_min}(K), K)$, P_{false_final} від L для різних значень K наведено на (Рисунок 2.11). Графіки показують характер поведінки оцінки (2.13) і демонструють наведені в (Таблиця 2.1) числові значення оцінки (2.13).

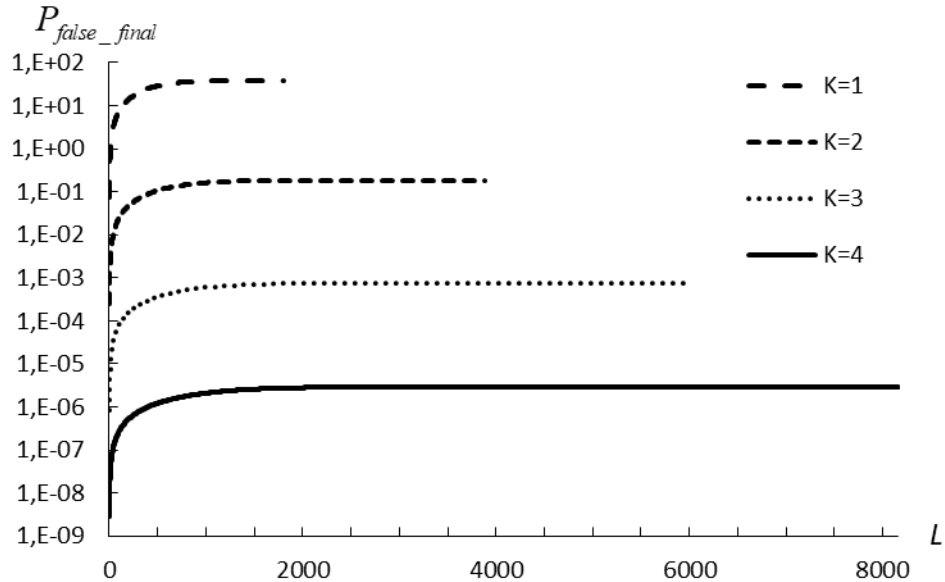


Рисунок 2.11 – Графіки залежності оцінки ймовірності хибної синхронізації від числа біт синхрокомбінації у ковзному вікні при $M = 8$ і $p_0 = 0.4$ для різних значень

З наведених результатів у (Таблиця 2.1) для отримання значень мінімальної ймовірності правильної синхронізації $P_{true_min} = 0.9997$ та максимальної ймовірності встановлення хибного синхронізму $P_{false_max} = 0.0003$, за параметрів $M = 8$, $p_0 = 0.4$, достатньо обрати значення величини $K = 4$.

Іншим потенційним способом зниження ймовірності хибної синхронізації може бути зменшення максимальної відстані d_{lim} до циклічних зсувів синхрокомбінації, під час ідентифікації уточненої послідовності R . Однак зі зменшенням значення d_{lim} це призводить до зменшення ймовірності правильної синхронізації, що обумовлює необхідність збільшення коефіцієнта

накопичення l . Цей підхід може бути ефективним, проте не розглядається в цій роботі.

2.5. Підхід до забезпечення ефективної синхронізації за відомою максимальною довжиною серії хибних спрацювань

Існує можливість використати алгоритм кадрової синхронізації без необхідності збільшення значення K . Така можливість реалізується шляхом визначення величини довжини серії випадків хибного спрацювання підсистеми кадрової синхронізації.

Випадки, коли підсистема синхронізації фіксує однакове положення межі синхрокомбінації для декількох поспіль зсувів бітів в ковзному вікні, утворюють *серію спрацювань* підсистеми синхронізації, а їхня *кількість визначає довжину цієї серії*.

Серією хибних спрацювань є ситуація, коли підсистема синхронізації фіксує однакові хибні положення межі синхрокомбінації для декількох поспіль зсувів бітів в ковзному вікні. Кількість таких зсувів є довжиною серії хибних спрацювань.

Підхід до забезпечення ефективної синхронізації за серією спрацювань підсистеми синхронізації полягає у:

- використанні ковзного вікна з фіксованою довжиною відповідно до максимальної довжини блоку синхрокомбінацій;
- аналізу довжини серії спрацювань підсистеми синхронізації;
- прийнятті рішення про встановлення синхронізму після досягнення довжини серії спрацювань підсистеми синхронізації певного порогового значення.

Водночас, процедури мажоритарної та кореляційної обробки в ковзному вікні з l прийнятих фрагментів залишаються незмінними.

Порогове значення довжини серії спрацювань підсистеми синхронізації обумовлюється розподілом довжин серій хибних спрацювань, яке може

визначатися шляхом моделювання, математичних розрахунків або інших доступних методів.

У цій роботі:

- розподіл довжин серій хибних спрацювань визначено моделюванням;
- пороговим значенням довжини серії спрацювань підсистеми синхронізації прийнято значення, на одиницю більше за максимальну довжину з ненульовою частотою в експериментально визначеному розподілі довжин серій хибних спрацювань.

2.6. Опис методу

Метод кадрової синхронізації на основі ковзного вікна з урахуванням серії спрацювань підсистеми синхронізації полягає в наступному.

1. Передавач ініціює передавання даних. Для цього передавач видає в канал синхрокомбінацію з l фрагментів. Фрагмент синхрокомбінації представлений визначеною перестановкою π , у якій мінімальна відстань Хеммінга d до всіх своїх циклічних зсувів є максимальною. Послідовність з l фрагментів формує блок синхронізації L_{block} , який після представлення синхрокомбінації-перестановки в двійковому вигляді шляхом рівномірного кодування його символів має довжину L біт.
2. Сторона приймача використовує ковзне вікно фіксованої довжини L біт, лічильник спрацювань підсистеми синхронізації та значення порогової довжини серії спрацювань підсистеми синхронізації. Приймач заповнює ковзне вікно бітами прийнятих фрагментів. Заповнення ковзного вікна виконується за схемою черги – FIFO (first in, first out).
3. Після заповнення ковзного вікна виконується мажоритарна обробка накопичених у ньому фрагментів, у результаті чого формується уточнена послідовність R .

4. Виконується кореляційна обробка послідовності R : визначається відстань Хеммінга від уточненої послідовності R до всіх циклічних зсувів синхрокомбінації π . Якщо для певного зсуву ця відстань менша або дорівнює $d_{\lim} = \left\lfloor \frac{d-1}{2} \right\rfloor$, то уточнена послідовність R ототожнюється з цим циклічним зсувом.
5. За відсутності ототожнення уточненої послідовності R з циклічним зсувом синхрокомбінації π , лічильник спрацювань підсистеми синхронізації скидається, і процес накопичення бітів у ковзному вікні продовжується з пункту 2 цього списку.
6. У разі ототожнення уточненої послідовності з циклічним зсувом синхрокомбінації π , що відповідає циклічному зсуву синхрокомбінації π , отриманому на попередньому етапі, приймач фіксує наявність синхронізації. Водночас він інкрементує лічильник спрацювань підсистеми синхронізації і процес синхронізації продовжується з пункту 2 цього списку.
7. За перевищення значення лічильника спрацювань підсистеми синхронізації порогового значення довжини серії спрацювань приймач приймає рішення про встановлення синхронізації. Виконує компенсацію фазового зсуву до межі фрагментів у ковзному вікні.
8. Після виконаної фазової компенсації приймач переходить до прийому та декодування наступного фрагменту інформації.

Таким чином використання ковзного вікна фіксованого розміру, мажоритарного та кореляційного аналізу прийнятих фрагментів дозволяє надійно встановити синхронізацію за фрагментами блоку синхронізації.

2.7. Алгоритми та імітаційна модель підсистеми кадрової синхронізації

З метою підвищення точності розпізнавання перестановки та забезпечення надійної синхронізації між передавачем і приймачем в умовах високої інтенсивності шуму в каналі зв'язку застосовано метод кадрової

синхронізації на основі ковзного вікна та серії спрацювань підсистеми синхронізації.

Для оцінювання ефективності та працездатності методу побудовано та використано імітаційну модель інформаційного обміну, яка працює у симплексному каналі передавання що використовує перестановки як формат подання даних.

2.7.1. Процедура інформаційного обміну та схема його реалізації

Передавач і приймач інформації з'єднані між собою каналом зв'язку. Передавання даних стаціонарним каналом зв'язку (Ch) відбувається в одну сторону, від однієї системи до іншої (симплексний тип передавання). Для кращого розуміння зазначено назви систем відповідно до їх цільової функції: перша система – передавання, відповідно: Tx , друга система – приймання, відповідно: Rx .

Передавання інформації відбувається від Tx через Ch до Rx . Канал зв'язку спотворює переданий сигнал шляхом накладання шуму (Noise) на вхідний сигнал, що призводить до бітової помилки з імовірністю p_0 . Відповідно, Rx приймає спотворений каналом зв'язку сигнал. Графічне представлення схеми інформаційного обміну наведено на (Рисунок 2.12).

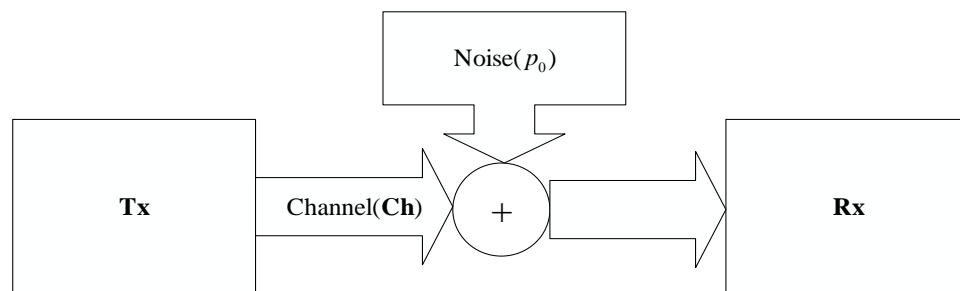


Рисунок 2.12 – Схема інформаційного обміну

На часовій діаграмі процедури передавання (Рисунок 2.13) зображено компоненти: Tx , Ch , Rx . Крім того, присутні фрагменти інформації, що знаходиться на виході Tx , вході Rx , вході і виході Ch .

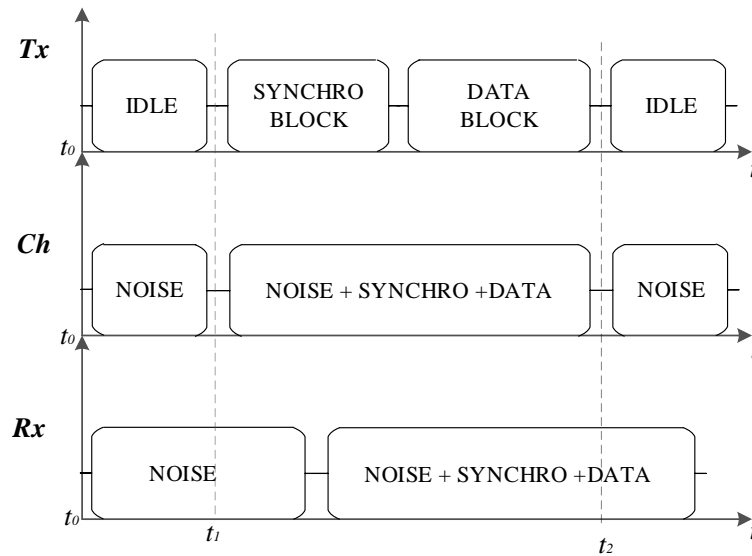


Рисунок 2.13 – Діаграма процедури передавання

У інтервалі часу $[t_0; t_1]$ Tx не надсилає жодні дані, тому рівень сигналу на вході каналу Ch залишається сталим і нульовим. Приймач нічого не «знає» про момент початку/завершення передавання, він постійно обробляє вхідний сигнал. У такому випадку в часовому проміжку $[t_0; t_1]$ Rx приймає тільки Noise. Відповідно, в цей період часу ймовірність бітової помилки в Rx дорівнює 0,5. У інтервалі часу $[t_1; t_2]$ передавач формує блок синхронізації та блок даних і послідовно видає їх в канал.

У процедурі передавання (Рисунок 2.13) приймач Rx постійно аналізує сигнал, що надходить з каналу зв'язку, оскільки додаткової інформації про початок передавання від Tx приймач не має, а процес передавання Tx може розпочати в будь-який момент часу (асинхронно). Відповідно, постає перша задача для приймача: синхронізація з передавачем для коректного визначення меж блоків, що передаються. Другою задачею є зменшення впливу шуму в каналі зв'язку на ефективність процедур передавання. Вирішення цих задач надає можливість реалізувати надійний інформаційний обмін у каналах зв'язку з високою ймовірністю бітової помилки.

2.7.2. Структура блоку синхронізації

На основі методу, викладеному в підрозділі (2.5), визначено структуру блоку синхронізації та блоку даних, що показано на (Рисунок 2.14). В контексті подальшого опису інформаційного обміну синхрокомбінація (обрана перестановка, що задовольняє вимогам) названа як синхролітера

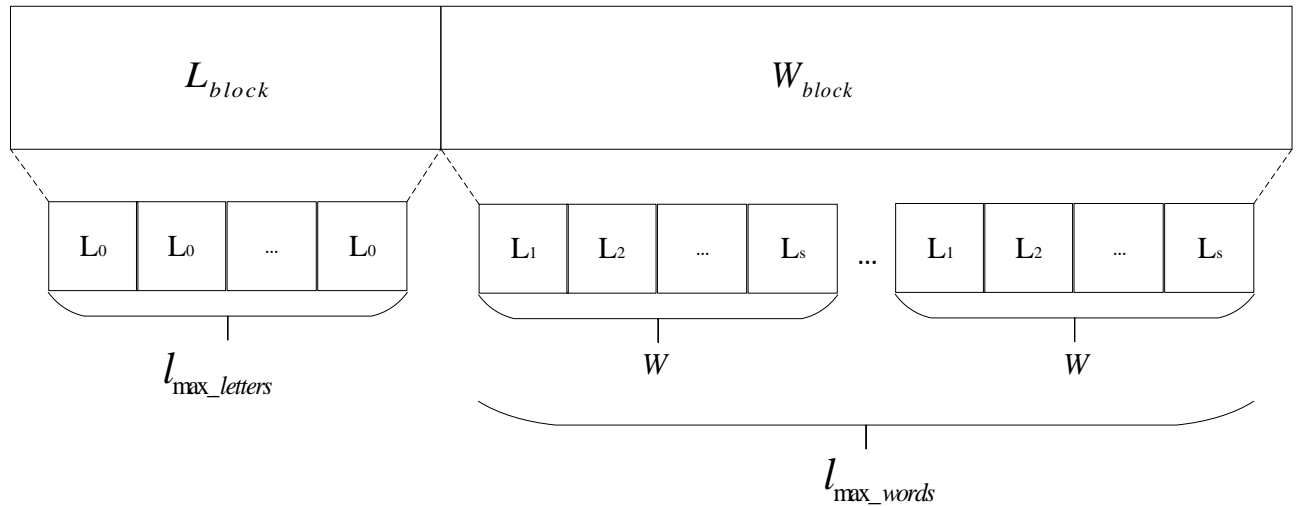


Рисунок 2.14 – Структура блоку синхронізації та блоку даних

На рисунку 2.14 позначено:

L_0 – синхролітера, визначена перестановкою π , що має максимальне значення мінімальної відстані Хеммінга d_{lim} до всіх своїх циклічних зсувів;

L_{block} – блок з $l_{max_letters}$ синхролітер L_0 (блок синхронізації);

W – слово, що складене з S літер – циклічних зсувів синхролітери: $\{L_1, L_2, \dots, L_s\}$;

W_{block} – блок з l_{max_words} слів W (блок даних – кадр).

Кожна літера L_j є перестановкою довжини M символів. Кожен символ закодований двійковим рівномірним кодом. Таким чином, довжина кожного символу дорівнює $l_r = \lceil \log_2 M \rceil$ біт. Відповідно, кількість біт у літері становить $n = M \cdot l_r$ біт, кількість біт у блоці синхронізації становить L_{block} :

$n_L = l_{\max_letters} \cdot M \cdot l_r$ біт, а кількість біт у блоці даних (кадрі) W_{block} становить:

$$n_W = l_{\max_words} \cdot M \cdot l_r \cdot (n - 1) \text{ біт.}$$

2.7.3. Довжина серії хибних спрацювань

У цій праці для отримання довжин серії хибних спрацювань підсистеми синхронізації виконано моделювання інформаційного обміну за визначеними параметрами. У інформаційному обміні беруть участь канал передавання, приймач та передавач. Моделювання дозволяє встановити максимальне значення довжини серії хибних спрацювань за заданих умов інформаційного обміну та визначити порогове значення довжини серії спрацювань підсистеми синхронізації.

Імітаційна модель передбачає проведення N випробувань. Алгоритм визначення розподілу довжин серій хибних спрацювань полягає в наступному:

1. Обирається перестановка-синхрокомбінація π , у якій значення мінімальної відстані Хеммінга від її двійкового представлення до всіх своїх циклічних зсувів є максимальним.
2. Для попередньо визначеної ймовірності бітової помилки p_0 обирається значення кількості ($l_{\max_letters}$) фрагментів синхрокомбінації, які складають блок синхронізації. Довжина блоку синхронізації в бітах n_L .
3. Фіксується початкове значення положення бітів в ковзному вікні n_{shift} як 0-вий біт.
4. На початку роботи моделі ковзне вікно приймача фіксованої довжини n_L біт заповнене шумом (бітами шуму з заданою ймовірністю бітової помилки каналу зв'язку), довжина ковзного вікна дорівнює довжині блоку синхронізації.
5. Передавач передає блок синхронізації каналом зв'язку.
6. Приймач отримує біти блоку синхронізації в ковзному вікні, виконує процедуру пошуку синхронізму, визначаючи межу синхрокомбінації. У

процесі прийому бітів з каналу ковзне вікно поступово заповнюється відповідно до діаграми на рисунку 2.13.

7. На кожному етапі зсуву бітів у ковзному вікні визначається уточнена послідовність R . Визначається відстань Хеммінга від уточненої послідовності R до всіх циклічних зсувів синхрокомбінації π . Якщо для певного зсуву ця відстань менша або дорівнює $d_{\lim} = \left\lfloor \frac{d-1}{2} \right\rfloor$, то уточнена послідовність R ототожнюється з цим циклічним зсувом.
8. Визначене ототожнене значення зсуву порівнюється з розрахунковим значенням зсуву від початку положення бітів у ковзному вікні (0-вий біт). Умовою перевірки хибного встановлення межі підсистемою кадрової синхронізації є наступний вираз (2.15).

$$S_{false} \rightarrow j_R \neq |n_{shift}|_n \quad (2.15)$$

де: j_R – ототожнений циклічний зсув (межа) уточненої послідовності R ;
 n_{shift} – кількість зсувів ковзного вікна.

9. Якщо приймач визначив межу, яка не еквівалентна розрахунковому значенню і ця межа ідентична попередній визначеній межі, значення довжини серії хибних спрацювань інкрементується. У іншому випадку лічильник довжини серії хибних спрацювань скидається.
10. Умовою завершення випробування є виконання всіх бітових зсувів у ковзному вікні приймача, яка дорівнює довжині ковзного вікна n_L . Зібрані результати (визначена межа – кількість бітів до наступного фрагменту, та довжина серії хибних спрацювань) для кожного з випробувань зводяться у зручний вигляд для аналізу.
11. У результаті отримується вибірка з значеннями довжин серій хибних спрацювань. У цій вибірці визначається максимальне значення довжини серії хибного визначення межі синхрокомбінації, яке використовується для подальшого встановлення мінімального порогу серії спрацювань підсистеми синхронізації.

Виконаємо застосування запропонованого алгоритму визначення довжин серії хибних спрацювань підсистеми синхронізації для таких параметрів:

- синхрокомбінація $\pi = (0, 1, 7, 3, 2, 5, 4, 6)$;
- довжина синхрокомбінації-перестановки $M = 8$;
- кількість біт для кодування символу перестановки $l_r = 3$;
- імовірність бітової помилки в каналі зв'язку за відсутності сигналу передавача $p_{noise} = 0.5$;
- імовірність бітової помилки за наявності сигналу передавача $p_0 = 0.4$;
- кількість випробувань у експерименті $N = 10000$;
- довжина фрагменту з синхролітерами у бітах –
 $n_L = l \cdot M \cdot l_r = 75 \cdot 8 \cdot 3 = 1800$ біт.

Сформовані результати зведені до гістограми абсолютних частот хибних спрацювань за заданими параметрами моделювання (Рисунок 2.15). На рисунку 2.15 представлено гістограму абсолютних частот (*absolute frequency*) довжин серій хибних спрацювань (*length*), які спостерігаються в результаті проведення експерименту з 10000 випробувань.

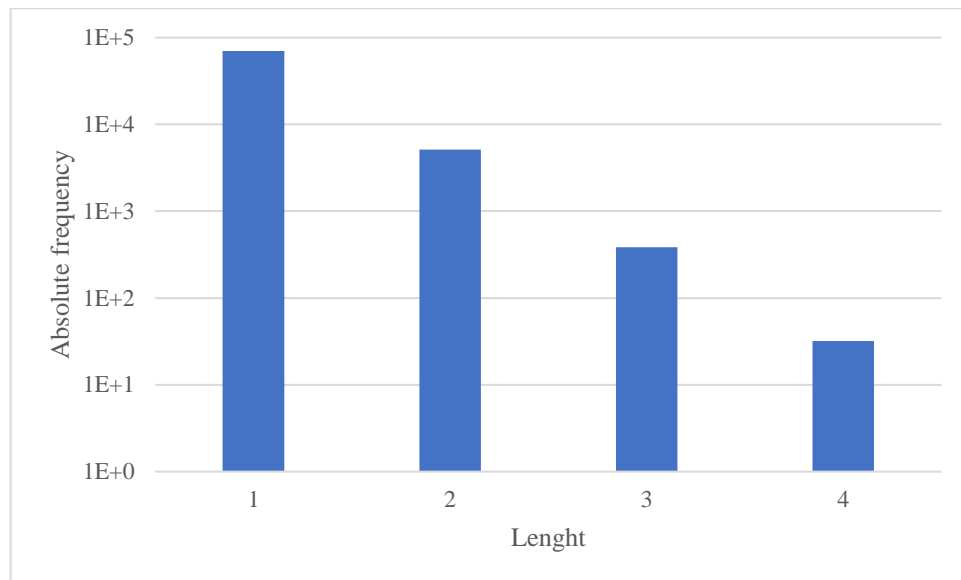


Рисунок 2.15 – Гістограма абсолютних частот серій хибних спрацювань

для $p_0 = 0.4$ і $l_{\max_letters} = 75$

З отриманих результатів імітаційного моделювання процесу синхронізації з визначеними вище параметрами моделювання значення максимальної довжини серії хибних спрацювань підсистеми синхронізації становить $l_{false_synch} = 4$. Отримане значення довжини серії хибних спрацювань дозволяє задати мінімальний поріг серії спрацювань підсистеми синхронізації під час приймання потоку бітів приймачем Rx в умовах високої інтенсивності шуму в каналі зв'язку.

Рішення про наявність синхронізму приймач приймає відповідно до пункту 7 описаного методу в розділі 2.6. Якщо лічильник серії хибних спрацювань підсистеми кадрової синхронізації перевищує заданий поріг довжини серії хибних спрацювань підсистеми синхронізації, приймається рішення про встановлення синхронізації. Після цього відбувається процес корегування фази (зміщення до початку межі наступного фрагменту). З визначеної позиції наступного фрагменту розпочинається спроба декодувати наступний фрагмент інформації, наприклад блок слів W_{block} .

За такої процедури синхронізації можлива ситуація, коли частина синхролітер L_0 блоку L_{block} може оброблятися приймачем як літери блоку

W_{block} (відбувається похибка в визначенні межі між блоками L_{block} і W_{block}). Чисельним значенням такої похибки вважається відстань від визначеної підсистемою синхронізації межі між блоками L_{block} і W_{block} до істинної межі. Ця відстань вимірюється в бітах.

На основі цих очікувань сформовано дві гіпотези та проведено імітаційне моделювання з метою підтвердження або спростування їх.

Гіпотеза 1. У результаті виконання процедури кадрової синхронізації за синхролітерами похибка в визначенні межі блоків L_{block} і W_{block} є (значення похибки не дорівнює нулю).

Гіпотеза 2. У результаті виконання процедури кадрової синхронізації за синхролітерами похибки в визначенні межі блоків L_{block} і W_{block} немає (значення похибки дорівнює нулю).

2.7.4. Імітаційна модель системи інформаційного обміну на основі перестановок

До структури передавача Tx (Рисунок 2.16) входять:

- підсистема формування блоку слів W_{block} , що отримує на вхід дані в вигляді перестановки π ;
- налаштування передавача за параметрами L_0 , $l_{max_letters}$, l_{max_words} , M задаються користувачем;
- підсистема формування блоку L_{block} формує вміст блоку відповідно до заданих параметрів передавача;
- підсистема видачі в канал зв'язку формує блоки L_{block} і W_{block} за встановленими параметрами передавання.

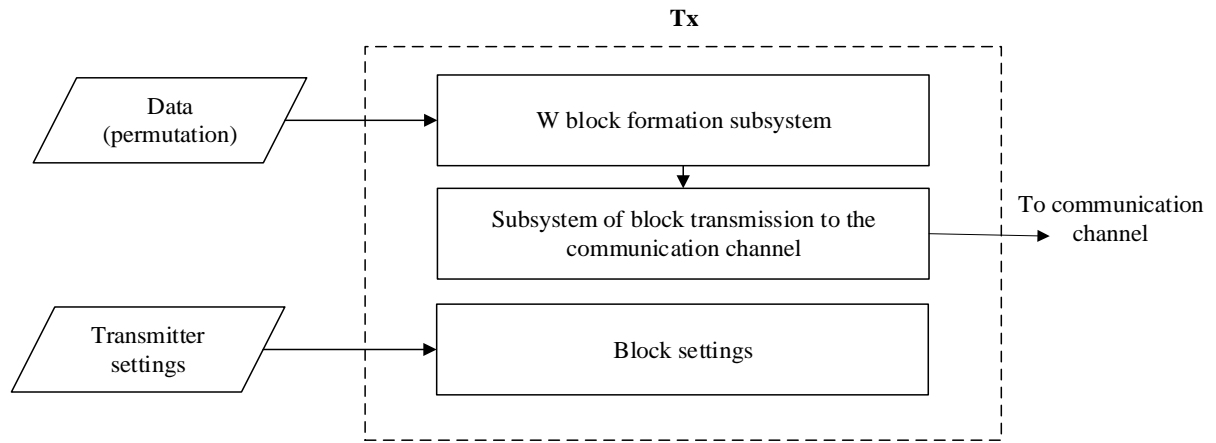


Рисунок 2.16 – Структура передавача Tx

Структура блоку каналу зв'язку *Ch* (Рисунок 2.17) містить:

- вхід, канал приймає дані (бітовий потік) та передає їх на блок накладання шуму;
- блок накладання шуму додає згенерований шум до вхідних бітів і передає на вихід;
- блок формування шуму, генерує спотворений сигнал відповідно до налаштування величини ймовірності помилкових бітів, що задається користувачем у вигляді значення p_0 . Сформований шум передається до блоку накладання шуму;
- вихід, отримує біти від блоку накладання шуму і передає на вихід каналу.

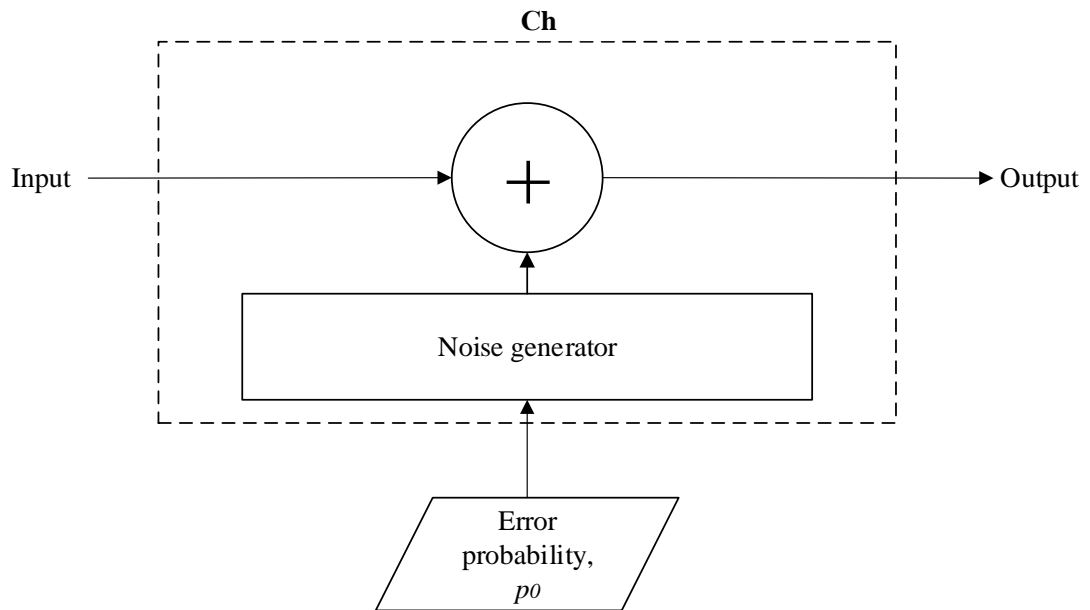


Рисунок 2.17 – Структура блоку каналу зв'язку Ch

Структура блоку приймача Rx (Рисунок 2.18) містить:

- вхід, приймає біти, що надійшли від системи каналу зв'язку Ch ;
- буфер вхідних бітів, накопичує інформацію з входу;
- лічильник кількості випадків синхронізації, фіксує довжину серії синхронізації;
- підсистема детектування синхронізації виконує аналіз бітів в буфері вхідних даних, результати аналізу відображаються шляхом збільшення або скидання лічильника кількості випадків спрацювання підсистеми синхронізації;
- підсистема фазової компенсації, виконує компенсацію фазового зсуву у випадку перевищення порогу серії підсистеми детектування синхронізації;
- підсистема обробки слова, аналізує буфер вхідних бітів після відпрацювання підсистемою фазової компенсації;
- буфер вихідних бітів, зберігає та відображає декодовану інформацію підсистемою обробки слова.

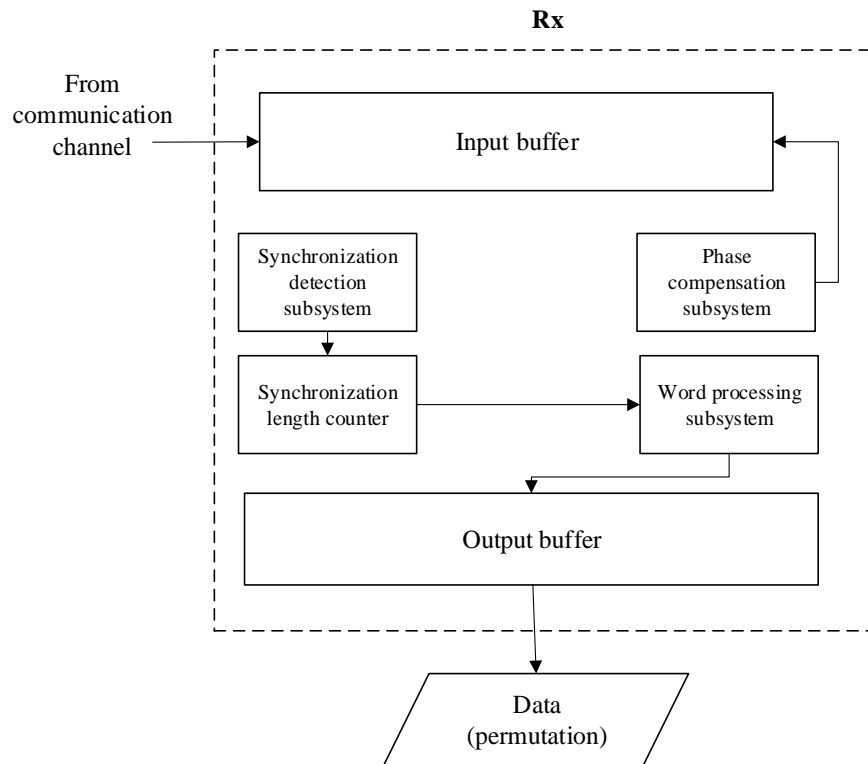


Рисунок 2.18 – Структура блоку приймача Rx

2.7.5. Алгоритми роботи компонентів системи інформаційного обміну на основі перестановок

Блок-схема алгоритму роботи блоку передавача (Рисунок 2.19) описує таку послідовність дій.

1. Ініціалізація. Налаштовує Tx на основі введених налаштувань:
 - довжина перестановки M ;
 - кількість біт на символ перестановки l_r ;
 - значення синхролітери L_0 у вигляді перестановки.
2. Перетворення перестановки π у двійкове представлення. Визначення розміру інформаційних векторів блоків синхронізації та даних відповідно до параметрів $l_{\max_letters}$ та l_{\max_words} .
3. Формування блоків L_{block} , W_{block} з перестановок у двійковому вигляді шляхом об'єднання в один вектор масив.

4. Видача блоків на вихід. Видає бітовий вміст блоків на вихід передавача, до каналу зв'язку. Побітове передавання відбувається поки весь вміст блоків не буде переданий на вихід.

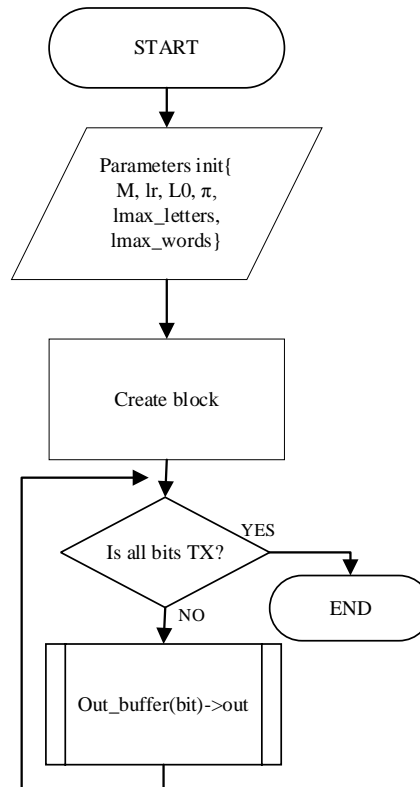


Рисунок 2.19 – Блок-схема алгоритму передавача Tx

Блок накладання шуму реалізує модель двійкового симетричного каналу зв'язку (binary symmetric channel, BSC). Двійковий симетричний канал передавання має: вхід (input) та вихід (output) з алфавітом двійкових значень, що приймають значення $\{1, 0\}$ (Рисунок 2.20). Вихід має імовірність помилкового оберненого символу, що надійшов на вхід p . Величина $1 - p$ задає імовірність передавання вхідного символу без змін, тобто без спотворення.

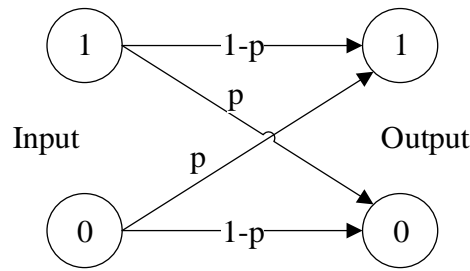


Рисунок 2.20 – Модель двійкового симетричного каналу передавання даних

Алгоритм роботи блоку приймача в наступному:

1. Ініціалізація. Налаштовує приймач відповідно до налаштувань:
 - значення синхролітери L_0 у вигляді перестановки π ;
 - величина довжини перестановки M ;
 - кількість біт на символ перестановки l_r ;
 - довжина перестановки в бітах n ;
 - значення всіх бітових, циклічних зсувів синхролітери L_0 ;
 - перетворює перестановку π у двійкове представлення;
 - кількість синхролітер $l_{\max_letters}$
 - кількість слів l_{\max_words} ;
 - мінімальне значення порогу серій синхронізації: $l_{synch_tresh} > l_{false_synch}$;
 - визначає розмір та вхідного та вихідного буферів на основі параметрів $l_{\max_letters}$ та l_{\max_words} .
2. Накопичення інформації з каналу. Виконання постійного накопичення бітів, що надходять на вхід блоку приймача та зберігаються у буфері вхідних бітів. Буфер має розмір, що дорівнює розміру блоків синхронізації та даних: $l_{rx_in} = L + W$ біт. Алгоритм запису в буфер при його заповненні можна представити у вигляді черги, перший прийшов - перший пішов (first in first out – FIFO).
3. Аналіз синхронізації. Накопичивши l_{rx_in} біт, приймач проводить аналіз частини вхідного буфера, яка дорівнює L біт, та формує уточнену

- послідовність R . Якщо уточнена послідовність біт асоціюється з L_0 , інкрементується лічильник серій синхронізації. Інакше, лічильник серій синхронізації скидається до нуля та виконується перехід на прийом наступного біту з входу блоку приймача (пункт 2).
4. Аналіз серії синхронізацій. При перевищенні значення мінімального порогу серії синхронізації l_{synch_tresh} , перехід до пункту 5, інакше прийом наступного біту (пункт 2).
 5. Фазова корекція. На основі отриманих даних синхролітери L_0 визначається величина фазового зсуву синхронізації. Для цього уточнена послідовність R порівнюється з усіма циклічними зсувами синхролітери $R \in [L_0; L_j]$. За наявності d_{lim} з певним зсувом отримується номер цього зсуву j . Вираховується кількість бітів до межі наступного блоку перестановки (фрагменту) $J_{shift} = n - j$. У разі значення $J_{shift} = 0$ перехід до пункту 6, якщо $J_{shift} \neq 0$, виконується накопичення кількості необхідних біт J_{shift} (пункт 2-5).
 6. Аналіз блоку слів. Виконується вибірка фрагменту даних з вхідного буферу, яка дорівнює W біт. Визначаються уточнені послідовності для кожної з $[L_1; L_j]$ літери в блоці W_{block} . Виконується порівняння з усіма циклічними зсувами синхролітери $R \in [L_1; L_j]$. У результаті порівняння отримується ототожнений номеру зсуву j для кожної з уточнених послідовностей. Отриманий номер зсуву записується у вихідний буфер. У випадку не розпізнавання слова після процедури порівняння з циклічними зсувами L_0 , вихідний буфер очищується і виконується перехід на пункт 2.
 7. Аналіз вмісту слова. Виконується підрахунок набраних номерів зсувів у вихідному буфері. При накопиченні $j = [1; n - 1]$ номерів зсувів, виконується перевірка, чи номери не повторюються (номери зсувів слів формують перестановку). Якщо номери є унікальними, у буфері

сформована перестановка, аналізується вміст буферу за індексом один. Якщо перший номер у вихідному буфері дорівнює одиниці (ототожнений з першим циклічним зсувом синхролітери-перестановки L_1) відбувається видача вмісту вихідного буфера користувачу. Формується висновок, що прийом даних (переданої перестановки передавачем) завершено. Якщо ж у буфері не були набрані унікальні значення номерів зсувів j або перший номер у вихідному буфері не дорівнює одиниці, то виконується очищення вихідного буферу і цикл прийому повторюється з пункту 2.

Блок-схема описаного алгоритму роботи приймача наведено на рисунку 2.21.

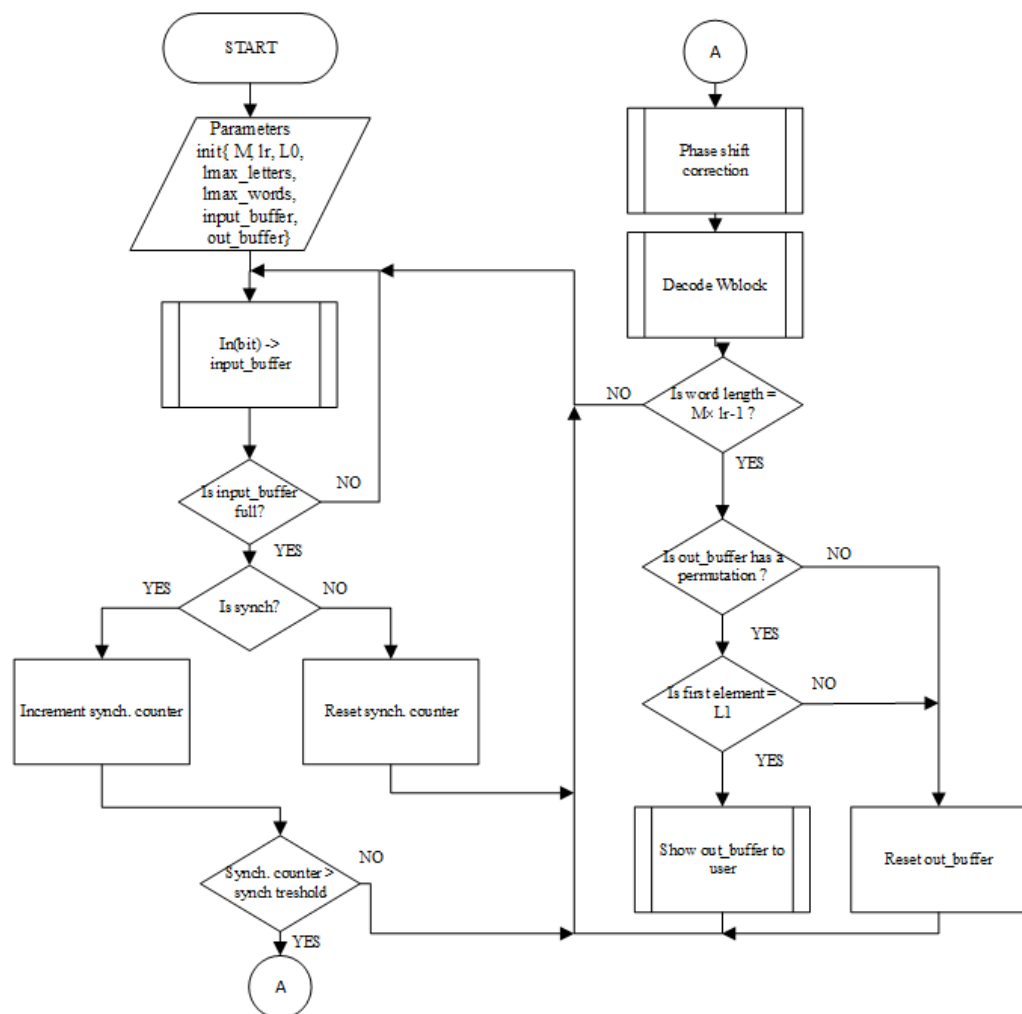


Рисунок 2.21 – Блок-схема алгоритму роботи блоку приймача R_x

2.7.6. Результати роботи імітаційної моделі

У процесі тестування імітаційну модель експериментально досліджено шляхом проведення 1000 експериментів. У ході експерименту передавались дані представлені у вигляді перестановки $\pi = (1, 2, \dots, 23)$ для різної імовірності бітової помилки в каналі зв'язку $p_0 = [0.1; 0.4]$ з кроком 0.05. Спочатку в канал зв'язку передавались L біт нульових значень, потім сформований блок синхронізації L_{block} та блок слів W_{block} . Параметри налаштування системи передавача:

- довжина перестановки $M = 8$, символів;
- кількість біт на символ перестановки $l_r = 3$ біт;
- значення синхролітери-перестановки $L_0 = (0, 1, 7, 3, 2, 5, 6)$;
- кількість блоків синхролітер $K = 1$;
- кількість синхролітер в блоці $l_{max_letters} = 75$;
- кількість слів у блоці даних $l_{max_words} = 89$.

Параметри налаштування системи приймача:

- довжина перестановки $M = 8$, символів;
- кількість біт на символ перестановки $l_r = 3$ біт;
- значення синхролітери-перестановки $L_0 = (0, 1, 7, 3, 2, 5, 6)$;
- кількість блоків синхролітер $K = 1$;
- кількість синхролітер в блоці $l_{max_letters} = 75$;
- кількість слів у блоці даних $l_{max_words} = 89$;
- мінімальне значення порогу серій синхронізації l_{synch_tresh} . У розділі 2.7.3 значення $l_{false_synch} = 4$, тому визначено $l_{synch_tresh} = 5$

За результатами проведеного експериментального дослідження імітаційної моделі сформована вибірка таких параметрів:

1. Кількість успішно прийнятих приймачем блоків даних $N_{frame_received}$ – ситуація, коли приймач зміг правильно виконати синхронізацію за літерами та декодував блок слів – випробування проведено успішно.

2. Кількість не прийнятих приймачем блоків даних N_{frame_lost} – ситуація, коли приймач не зміг правильно виконати синхронізацію за літерами або правильно виконав синхронізацію, проте неправильно декодував блок слів – випробування не проведено успішно.
3. Максимальна кількість біт, що потрапила з блоку L_{block} до ковзного вікна аналізу блоку слів W_{block} , – $l_{max_letter_block}$
4. Мінімальна кількість біт, що потрапила з блоку L_{block} до ковзного вікна аналізу блоку слів до W_{block} , – $l_{min_letter_block}$.
5. Кількість успішно декодованих слів з максимальною кількістю бітів, що потрапили з блоку L_{block} до ковзного вікна аналізу блоку W_{block} , – $N_{rx_max_l}$.
6. Кількість успішно декодованих слів з мінімальною кількістю бітів, що потрапили з блоку L_{block} до W_{block} , – $N_{rx_min_l}$.
7. Середній час передавання t_{μ} в секундах.

Отримані результати для кожної ймовірності $p_0 = [0.1; 0.4]$ з кроком 0.05 у результаті 1000 випробувань наведено в (Таблиця 2.2).

Таблиця 2.2 – Результати експерименту

Імовірність бітової помилки, p_0	0.1	0.15	0.2	0.25	0.3	0.35	0.4
$N_{frame_received}$	1000	1000	1000	1000	1000	1000	1000
N_{frame_lost}	0	0	0	0	0	0	0
$l_{max_letter_block}$	552	552	552	552	552	552	552
$l_{min_letter_block}$	_*	_*	_*	_*	_*	_*	0
$N_{rx_max_l}$	1000	1000	1000	1000	1000	1000	988
$N_{rx_min_l}$	0	0	0	0	0	0	12
t_{μ}	0.716	0.706	0.689	0.7	0.7	0.67	0.666

Примітки: * - значення дорівнює максимальній кількості

Для оцінки впливу хибного визначення межі синхролітери підсистемою синхронізації приймача та перевірку ефективності алгоритму роботи приймача проведено ще 1000 випробувань, з аналогічними параметрами випробування, параметр мінімального значення порогу серії спрацювань синхронізації встановлено на рівні $l_{synch_tresh} = -1$ у блоці приймача Rx . За алгоритмом роботи блоку приймача (Рисунок 2.21) такий показник вимикає аналіз серій. Результати проведеного експерименту наведено в (Таблиця 2.3).

Таблиця 2.3 – Результати експериментів за вимкненого механізму аналізу серії спрацювань підсистеми синхронізації

Імовірність бітової помилки, p_0	0.1	0.15	0.2	0.25	0.3	0.35	0.4
$N_{frame_received}$	1000	1000	1000	1000	1000	1000	1000
N_{frame_lost}	0	0	0	0	0	0	0
$l_{max_letter_block}$	552	552	552	1104	1104	1104	1104
$l_{min_letter_block}$	_*	_*	_*	552	552	552	552
$N_{rx_max_l}$	1000	1000	1000	2	56	317	318
$N_{rx_min_l}$	0	0	0	998	944	683	682
t_μ	0.68	0.681	0.69	0.736	0.702	0.613	0.488

Примітки: * - значення дорівнює максимальній кількості

За результатами експерименту проведено 1000 випробувань (Таблиця 2.2). Втрачених переданих блоків у ході випробувань не зафіксовано, алгоритм прийому блоку приймача із запропонованою структурою (Рисунок 2.18) успішно прийняв всі передані фрагменти каналом зв'язку з імовірностями

бітової помилки $p_0 = [0.1; 0.4]$ з кроком 0.05. За цих умов, похибка у визначенні відстані між блоками L_{block} і W_{block} не перевищувала значення 552 біт. Для параметрів $M = 8$ і $l_r = 3$ це дорівнює одному слову (фазовий зсув за словами) W . Середній час синхронізації і декодування одного блоку слів становить 0,666 с за ймовірності бітової помилки каналу $p_0 = 0.4$. Відносна частота появи похибки визначення межі між блоками L_{block} і W_{block} на одне слово для $p_0 = 0.4$ дорівнює $\omega_{word} = \frac{988}{1000} = 0.988$. Відносна частота появи нульової похибки визначення межі між блоками L_{block} і W_{block} дорівнює $\omega_0 = 1 - \omega_{word} = 0.012$. За ймовірності бітової помилки каналу p_0 від 0.1 до 0.35 відносна частота появи похибки визначення межі між блоками L_{block} і W_{block} на одне слово становить 1.0.

Результати проведення 1000 випробувань наведено в (Таблиця 2.3). Алгоритм блоку приймача виконував аналіз досягнення порогового значення серії спрацювань підсистеми синхронізації за параметром l_{synch_tresh} . За результатами в (Таблиця 2.3) втрачених переданих фрагментів у ході випробувань не зафіксовано. Значення похибки визначення межі між блоками L_{block} і W_{block} становить 1104 біти, що дорівнює двом словам $2 \cdot W$ за ймовірності бітових помилок у каналі p_0 , починаючи з 0.25 до 0.4. Відносні частоти значень похибки визначення межі між блоками L_{block} і W_{block} показано на рисунку 2.22. Середній час синхронізації та декодування для одного випробування становить 0,488 с для $p_0 = 0.4$.

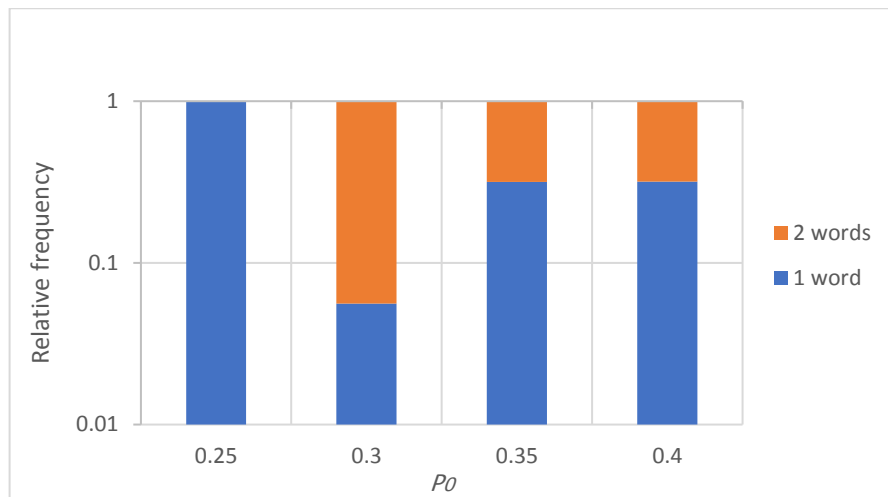


Рисунок 2.22 – Гістограма відносних частот значень похибки визначення межі між блоками L_{block} і W_{block}

2.7.7. Особливості імітаційної моделі

У наведеній імітаційній моделі системи інформаційного обміну використано як канал передавання модель двійкового симетричного каналу (BSC). Така модель може бути використана для теоретичного аналізу каналів передавання даних, зокрема в задачах обчислення пропускну здатності, як показано в роботі авторів [153], де розглянуто оптимізацію за допомогою симетризованої KL-інформації. Разом з тим, модель BSC є спрощеною ідеалізацією реальних каналів передавання даних, яка має низку недоліків у контексті імітаційного моделювання шуму:

1. Нереалістичне припущення рівномірного розподілу помилок. BSC передбачає однакову ймовірність помилки для кожного біта незалежно від попередніх. У реальних каналах помилки часто мають кореляцію.
2. Відсутність моделювання типу шуму. Модель BSC не враховує фізичну природу шуму (прикладом можуть бути білий Гауссівський шум, імпульсний шум або інтерференції), що є важливим в реальних системах.
3. Обмеження адаптації до середовища. BSC не враховує змінні характеристики каналу, такі, як динамічне відношення сигнал-шум (SNR), що може змінювати ймовірність помилки залежно від поточних

умов. Така характеристика притаманна в складних модемних системах де представлення і передавання інформації реалізується за допомогою фазової, амплітудної модуляції, де помилки залежать від взаємодії сигналу і шуму.

4. Ігнорування втрат пакетів або символів. У багатьох реальних каналах можуть виникати не лише помилки у бітових значеннях, але й втрати даних, що BSC не моделює.
5. Неврахування просторово-часових ефектів. BSC не враховує ефекти багатопрореневості та зміщення сигналу в часі, які можуть значно впливати на якість передавання.
6. Робота авторів [154] показує вплив просторово-часових ефектів на завадостійкість телекомунікаційних систем з orthogonal frequency-division multiplexing (OFDM) модуляцією. При реалізації імітаційної моделі системи інформаційного обміну явище зміщення сигналу в часі враховане шляхом реалізації передавача у вигляді окремої системи, а також передаванню блоку нульових бітових значень довжиною $L = l_{\text{max_letters}} \cdot M \cdot l_r$ в канал, що після проходження системи каналом зв'язку містить шум.

Альтернативою для використання BSC моделі у якості каналу зв'язку є:

1. Двійковий канал із залежною ймовірністю помилок. Для врахування кореляції помилок можливо використати моделі з марківськими процесами. Так, автори роботи [155] пропонують використання марківських моделей для кількісної оцінки надійності мереж, що дозволяє прогнозувати стани мережі та планувати превентивні заходи для забезпечення її стабільної роботи.
2. Моделювання реального шуму. Для точнішого відображення середовища використати канали, що враховують тип шуму: канал із білим гауссівським шумом Additive White Gaussian Noise (AWGN); канал із імпульсним шумом, для випадків, коли інтерференції є короткочасними, та інтенсивними. Робота авторів [156], де проводились

дослідження і оцінювання стандарту стільникової мережі 5G 3GPP, використала декілька моделей каналу передавання, серед них є як AWGN канал, так і BSC модель.

3. Моделі з багатопроменевістю. Наприклад, моделі Релея або Райса для моделювання багатопроменевих ефектів у бездротових мережах.
4. Канали із змінним відношенням сигнал-шум signal-to-noise ratio (SNR).
5. Ітеративне моделювання за допомогою реалістичних симуляторів. Використання програмного забезпечення MATLAB, огляд функціональних можливостей якого показано у виданні авторів [157]. У праці [158] розглянуто можливості застосування симулятору мереж NS-3 для дослідження LoRAWAN протоколу, що застосовується в IoT. Використання подібних інструментів дозволяє інтегрувати складніші моделі каналів із урахуванням реальних фізичних ефектів.

Для збереження простоти і збільшення реалістичності двійкового симетричного каналу передавання можна покращити деякі аспекти шляхом застосування моделі Гільберта-Елліота. Автори [159] розглядають вплив шуму та невизначеності на пропускну здатність повідомлень, використовуючи модель Гільберта-Елліота для моделювання каналу зв'язку. Описані підходи допоможуть підвищити точність імітаційних досліджень і забезпечити більшу відповідність реальним умовам передавання інформації.

При процедурі синхронізації та декодування слова, що використана в імітаційній моделі інформаційного обміну, компенсація фазового зсуву відбувається тільки до межі наступного фрагменту синхролітери, а не до початку блоку зі словами W_{block} , оскільки приймач не знає, де початок блоку в буфері вхідної інформації з каналу зв'язку Ch . Тому маючи значення синхролітери-перестановки її структуру та властивості, алгоритм блоку приймача в імітаційній моделі інформаційного обміну компенсує фазовий зсув ковзного вікна лише до початку наступного фрагменту перестановки. За хибного визначення меж блоку слова і блоку літер виникає похибка у визначенні межі, що впливає на імовірність бітової помилки в блоці слів.

Аналіз результатів, наведених у (Таблиця 2.2), за ймовірності бітової помилки в каналі зв'язку $p_0 = 0.4$ свідчить про наявність похибки у визначенні межі в 552 біти. Цей фрагмент не належить блоку даних і, відповідно, збільшує ймовірність бітової помилки в ньому. Оцінити цю ймовірність можна виразом (2.16):

$$p_0^* = \frac{l_{\max_letter_block} \cdot 0.5 + (n_w - l_{\max_letter_block}) \cdot p_0}{n_w} \quad (2.16)$$

Для визначених параметрів моделі $p_0^* = 0.401$. Похибка визначення межі виникає внаслідок того, що приймач не знає точного початку блоку синхролітер-перестановок і опирається лише на інформацію про структуру інформаційного потоку. Відповідно, після встановлення синхронізації система виконує корекцію фазового зсуву до межі літери і приймає припущення, що наступний блок з W біт може містити інформацію зі словами. Зменшення ймовірності бітової помилки при декодуванні слова залежить від точного визначення межі блоків L_{block} та W_{block} . Одним із засобів підвищення точності є використання методу кадрової синхронізації на основі ковзного вікна та серії спрацювань підсистеми синхронізації. Механізм встановленого порогу серії спрацювань підсистеми синхронізації представлено в вигляді лічильника, що показує довжину серії синхронізації, в алгоритмі моделі як l_{synch_tresh} .

Ефективність роботи методу кадрової синхронізації на основі ковзного вікна та серії спрацювань підсистеми синхронізації можливо оцінити шляхом порівняння результатів, представлених у (Таблиця 2.2 та Таблиця 2.3). За результатами двох експериментів можна зробити висновок про висунуті гіпотези 1 і 2 щодо механізму відстеження довжини серії спрацювань підсистеми синхронізації і заданого порогового значення серії спрацювань підсистеми спрацювань синхронізації. Гіпотеза 1 підтвердилась, оскільки після перевищення порогу серії спрацювань підсистеми синхронізації, корекції фазового зсуву, початку аналізу та декодуванню слова, величина похибки визначення між блоками L_{block} і W_{block} не є нульовою. Відповідно,

гіпотеза 2 не підтвердилась. Однак за розрахункової граничної ймовірності бітової помилки в каналі зв'язку $p_0 = 0.4$ для кількості блоків синхролітер $K = 1$ та $l_{\max_letters} = 75$ трапляються випадки, де похибка визначення межі блоків L_{block} і W_{block} дорівнює нулю, кількість таких випадків 12 з 1000 випробувань. Таблиця 2.2 демонструє результати за умов використання методу на основі ковзного вікна та серії спрацювань підсистеми синхронізації. У такому випадку максимальна величина похибки визначення межі блоків синхронізації та фрагментом даних становить 552 біти.

Натомість у результаті проведених випробувань, де запропонований метод не використовувався, величина похибки визначення межі блоків L_{block} і W_{block} становить 1104 біти (Таблиця 2.3). Відповідно, застосування методу на основі ковзного вікна та серії спрацювань підсистеми синхронізації допомагає підвищити точність визначення межі блоків L_{block} і W_{block} . Аналіз порогу серії спрацювань підсистеми синхронізації підсистеми синхронізації допоміг скоротити величину фазового зсуву за словами вдвічі та підвищив точність розпізнавання слова.

Середній час, що витрачений для декодування слова у випадку використання методу, становив 0.666 с за ймовірності бітової помилки в каналі $p_0 = 0.4$. За відсутності механізму детектування перевищення порогу серії спрацювань підсистеми синхронізації цей час за тієї ж ймовірності бітової помилки в каналі становив 0.488 с, що на 26.7% швидше. Ймовірність бітової помилки в блоці слів при аналізі та декодуванню при цьому збільшується з величини 0.401 до 0.402. Компенсувати похибку у визначенні межі блоків L_{block} і W_{block} можливо шляхом виконання компенсації визначеної межі після прийняття рішення про синхронізм на величину одного слова під час виконання аналізу та декодуванні блоком приймача. Таким чином похибка у визначенні межі блоків синхролітер і слів зменшиться з показника 99.8% до 0,2% за визначених параметрів моделі та ймовірності бітової помилки в каналі зв'язку $p_0 = 0.4$.

Відмінністю запропонованого алгоритму синхронізації від алгоритму в [103] є використання ковзного вікна фіксованого розміру та визначення мінімального порогу серії спрацювань серії синхронізацій. Як наслідок, запропонований алгоритм не використовує динамічну зміну значень K і l – кількості блоків (K) з l фрагментів довжиною M прийнятих з каналу символів. Відтак, значення K і l є сталими. Їх обирають таким чином, щоб задовольняти вимогам до ймовірностей правильної та хибної синхронізації. Довжина ковзного вікна дорівнює $K \cdot L \cdot M$ символів.

2.8. Висновки

У другому розділі розвинуто метод кадрової синхронізації систем передавання даних з нероздільним факторіальним кодуванням, який за рахунок використання ковзного вікна фіксованого розміру та врахування довжини серії спрацювань підсистеми синхронізації дозволяє знаходити межі синхрокомбінації за високої ймовірності бітової помилки в каналі зв'язку та невідомого початкового моменту передавання даних. Значення мінімального порогу серії спрацювань підсистеми синхронізації визначається через максимальну довжину серії хибних спрацювань підсистеми синхронізації.

Розроблено алгоритм кадрової синхронізації на основі ковзного вікна для симплексних систем зв'язку з невизначеним моментом початку прийому синхрокомбінації. Для цього отримано математичну оцінку ймовірності правильної синхронізації під час обробки фрагментів синхрокомбінації в ковзному вікні з фіксованим розміром. Запропоновано підхід для підвищення ймовірності правильної синхронізації без необхідності збільшення кількості блоків з фрагментами при використанні запропонованого методу.

Розроблено алгоритм роботи підсистеми кадрової синхронізації на основі ковзного вікна та серії спрацювань підсистеми синхронізації алгоритм визначення довжин серії хибних спрацювань підсистеми синхронізації, алгоритм обробки інформаційного блоку перестановок-слів у симплексних

системах з нероздільним факторіальним кодуванням. Визначено структуру інформаційного обміну та алгоритми взаємодії передавача, каналу зв'язку та приймача. Запропоновано структуру інформаційного фрагмента, що включає блок синхронізації (L_{block}) та блок інформаційних слів (W_{block}).

Проведене імітаційне моделювання (10000 випробувань) дозволило встановити максимальну довжину серії хибних спрацювань підсистеми синхронізації ($l_{false_synch} = 4$) для ймовірності бітової помилки $p_0 = 0.4$. Подальше тестування (1000 випробувань) підтвердило, що врахування цієї довжини зменшує похибку визначення меж блоків до 552 біт (1 слово) порівняно з 1104 бітами (2 слова) без її врахування.

Запропоновані методи та алгоритми підтвердили свою ефективність і можуть бути використані для реалізації кадрової синхронізації в системах завадостійкого інформаційного обміну на основі перестановок, зокрема у трьохетапному криптографічному протоколі.

Основні результати дослідження цього розділу представлено в [28], [160].

3. СКІНЧЕННІ ПОЛЯ КВАДРАТНИХ МАТРИЦЬ ПОРЯДКУ 2

3.1. Вступ

Комутативні операції відіграють базову роль у алгоритмах криптографічного перетворення інформації, широко застосовуваних у сучасних інформаційно-комунікаційних системах, смарттехнологіях, інтернеті речей. Зокрема, комутативні криптографічні перетворення використовують процедури узгодження ключів, асиметричне шифрування, трьохетапні криптографічні протоколи.

Одним з класичних криптографічних схем, які використовують комутативне перетворення піднесення до степеня в модульній арифметиці, є протокол Діффі — Геллмана [134], RSA [161], SRA [162], криптосистема Мессі — Омура [163].

Теорія скінченних полів також відіграє одну з ключових ролей у криптографії. Так, операції додавання, множення, пошуку обернених значень у симетричних алгоритмах шифрування [128], [133] реалізують над розширеним скінченним полем $GF(2^n)$. На теорії скінченних полів будують алгоритми перевірки на простоту та факторизації цілих чисел, що є фундаментом асиметричної криптографії [134], [135], [136].

Цей розділ спрямовано на виявлення та дослідження сімейств квадратних матриць з комутативною операцією множення. Матриці обмежено розмірністю 2×2 , а їх елементи належать простому полю лишків. А також розробці та дослідженню алгоритмів формування перестановок з використанням квадратних матриць розмірністю 2×2 з метою узгодження ключів-перестановок для факторіального кодування даних на основі відомої учасникам інформаційної взаємодії ключової матриці.

3.2. Комутативні сімейства матриць 2×2

Наведемо наступне означення.

Означення 1 [164], [165]. Повна (або загальна) лінійна група порядку n над будь-яким полем F або кільцем R – це група оборотних матриць $n \times n$ з елементами з F (або R), де груповою операцією є звичайне множення матриць.

Позначати повну загальну групу порядку n над полем F будемо через $GL(n, F)$.

Зазначимо, що квадратна матриця A є оборотною тоді і тільки тоді, коли її визначник $|A| \neq 0$ [166].

Розглянемо групу $\Gamma = \left\{ A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, a, b, c, d \in \mathbb{Z}_p, |A| \neq 0 \right\}$, де \mathbb{Z}_p – просте

поле лишків за модулем p . Тоді $\Gamma = GL(2, \mathbb{Z}_p)$.

Твердження 1. Операція множення є комутативною для наступних сімейств матриць групи $\Gamma = \left\{ A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, a, b, c, d \in \mathbb{Z}_p, |A| \neq 0 \right\}$:

$$\Gamma_1 = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}, t, a \in \mathbb{Z}_p, t \neq 0, a \neq 0 \right\},$$

$$\Gamma_2 = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ a & ak+1 \end{pmatrix}, t, a, k \in \mathbb{Z}_p, t \neq 0, ak+1 \neq 0 \right\}, k - \text{фіксоване};$$

$$\Gamma_3 = \left\{ t \cdot \begin{pmatrix} 1 & a \\ 0 & ak+1 \end{pmatrix}, t, a, k \in \mathbb{Z}_p, t \neq 0, ak+1 \neq 0 \right\}, k - \text{фіксоване};$$

$$\Gamma_4 = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & 0 \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}, a \text{ і } b - \text{фіксовані};$$

$$\Gamma_5 = \left\{ t \cdot \begin{pmatrix} 0 & 1 \\ b & a \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}, a \text{ і } b - \text{фіксовані};$$

$$\Gamma_6 = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, t, a, b, k \in \mathbb{Z}_p, t \neq 0, b \neq 0, a(a+k)-b \neq 0 \right\}, b \text{ і } k - \text{фіксовані}.$$

Доведення. Зауважимо, що в загальному випадку група Γ неабелева.

Розглянемо наступні випадки для $|A| \neq 0$:

- 1) $b = c = 0, ad \neq 0$;
- 2) $b = 0$ або $c = 0, ad \neq 0$;
- 3) $bc \neq 0, ad = 0$;
- 4) $ad \neq 0, bc \neq 0$.

1. Випадок 1: $b = c = 0$ і $ad \neq 0$. Тоді $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & d \end{pmatrix} = a \begin{pmatrix} 1 & 0 \\ 0 & d/a \end{pmatrix}$.

Множина таких діагональних невироджених матриць еквівалентна множині

$$\Gamma_1 = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}, t, a \in \mathbb{Z}_p, t \neq 0, a \neq 0 \right\}, \text{ яка утворює абелеву групу [167]: для}$$

$\forall A, B \in \Gamma_1$ справедливо $AB = BA$.

2. Випадок 2: $b = 0$ або $c = 0, ad \neq 0$.

Нехай $b = 0$ і $ad \neq 0$. Тоді матриця $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} a & 0 \\ c & d \end{pmatrix} = a \begin{pmatrix} 1 & 0 \\ c/a & d/a \end{pmatrix}$.

Множина таких матриць є сімейством невироджених нижньотрикутних

$$\text{матриць } \Upsilon = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ a & b \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}, \text{ яка відповідно до [167] утворює}$$

групу за множенням.

$$\text{Нехай } A, B \in \Upsilon \text{ і } A = \begin{pmatrix} 1 & 0 \\ a & b \end{pmatrix}, B = \begin{pmatrix} 1 & 0 \\ x & y \end{pmatrix}. \text{ Добуток } A \cdot B = \begin{pmatrix} 1 & 0 \\ a + bx & by \end{pmatrix}.$$

$$\text{Добуток } B \cdot A = \begin{pmatrix} 1 & 0 \\ x + ay & by \end{pmatrix}.$$

Значення $AB = BA$, якщо $a + bx = x + ay$ або $x(b-1) = a(y-1)$. Якщо $a = x = 0$, Υ вироджується в Γ_1 , яка є абелевою. Якщо ж $a, x \neq 0$, прийmemo

$$\frac{b-1}{a} = \frac{y-1}{x} = k, \quad k \in \mathbb{Z}_p. \quad \text{Тоді } \begin{cases} b = ak + 1; \\ y = xk + 1; \end{cases} \text{ а матриці } A = \begin{pmatrix} 1 & 0 \\ a & ak + 1 \end{pmatrix},$$

$$B = \begin{pmatrix} 1 & 0 \\ x & xk + 1 \end{pmatrix}, \text{ де } a \neq 0, x \neq 0, ak + 1 \neq 0, xk + 1 \neq 0, \forall k \in \mathbb{Z}_p.$$

Тоді $\Gamma_2 = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ a & ak+1 \end{pmatrix}, t, a, k \in \mathbb{Z}_p, t \neq 0, ak+1 \neq 0 \right\}$ є абелевою групою,

причому значення a і t можуть бути довільними, а значення k є фіксованим параметром групи Γ_2 .

Прийнявши $c=0$, $ad \neq 0$, за аналогією можна показати, що група невідроджених верхньотрикутних матриць виду $A = \begin{pmatrix} a & b \\ 0 & d \end{pmatrix}$ є абелевою, якщо

утворює групу $\Gamma_3 = \left\{ t \cdot \begin{pmatrix} 1 & a \\ 0 & ak+1 \end{pmatrix}, t, a, k \in \mathbb{Z}_p, t \neq 0, ak+1 \neq 0 \right\}$.

3. Випадок 3: $bc \neq 0$, $ad=0$.

Нехай $bc \neq 0$, $d=0$. Тоді матриця $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = \begin{pmatrix} a & b \\ c & 0 \end{pmatrix} = a \begin{pmatrix} a/b & 1 \\ c/b & 0 \end{pmatrix}$.

Такі матриці задають множину $\Xi = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & 0 \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}$.

Нехай $A, B \in \Xi$ і $A = t \begin{pmatrix} a & 1 \\ b & 0 \end{pmatrix}$, $B = s \begin{pmatrix} x & 1 \\ y & 0 \end{pmatrix}$. Рівність $A \cdot B = B \cdot A$ означає

$\begin{pmatrix} ax+y & a \\ bx & b \end{pmatrix} = \begin{pmatrix} ax+b & x \\ ay & y \end{pmatrix}$ або $\begin{cases} x=a; \\ y=b. \end{cases}$ Отже, для $bc \neq 0$ і $d=0$ комутативним

сімейством є множина $\Gamma_4 = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & 0 \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}$, де a і b –

фіксовані.

Прийнявши $bc \neq 0$, $a=0$, за аналогією можна показати, що комутативним сімейством є множина $\Gamma_5 = \left\{ t \cdot \begin{pmatrix} 0 & 1 \\ b & a \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}$,

де a і b – фіксовані.

Зауважимо, що сімейства Γ_4 , Γ_5 не замкнено відносно операції множення, тому не утворюють групу.

4. Випадок 4: $\begin{cases} ad \neq 0; \\ bc \neq 0. \end{cases}$ Оскільки $b \neq 0$, то $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = b \begin{pmatrix} a/b & 1 \\ c/b & d/b \end{pmatrix}$.

Множина таких матриць утворює множину невироджених матриць

$$\Psi = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & c \end{pmatrix}, t, a, b, c \in \mathbb{Z}_p, t \neq 0, b \neq 0, ac - b \neq 0 \right\}.$$

Нехай $A, B \in \Psi$ і $A = \begin{pmatrix} a & 1 \\ b & c \end{pmatrix}$, $B = \begin{pmatrix} x & 1 \\ y & z \end{pmatrix}$. Добуток

$$A \cdot B = \begin{pmatrix} ax + y & a + z \\ bx + cy & b + cz \end{pmatrix}, \quad B \cdot A = \begin{pmatrix} ax + b & x + c \\ ay + bz & y + cz \end{pmatrix}.$$

Добуток

Рівність $AB = BA$ досягається, якщо $\begin{cases} ax + y = ax + b; \\ a + z = x + c; \\ bx + cy = ay + bz; \\ b + cz = y + cz. \end{cases}$ З цього слідує, що

$$\begin{cases} y = b; \\ c - a = z - x. \end{cases}$$

Нехай $c - a = z - x = k$, $k \in \mathbb{Z}_p$. Тоді $A = \begin{pmatrix} a & 1 \\ b & a + k \end{pmatrix}$, $B = \begin{pmatrix} x & 1 \\ b & x + k \end{pmatrix}$,

звідки комутативним сімейством є множина

$$\Gamma_6 = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a + k \end{pmatrix}, t, a, b, k \in \mathbb{Z}_p, t \neq 0, b \neq 0, a(a + k) - b \neq 0 \right\},$$

причому значення a і t можуть бути довільними, а значення b і k є фіксованими параметрами множини Γ_6 .

Запишемо всі сімейства $\Gamma_1 - \Gamma_6$ матриць з $\Gamma = GL(2, \mathbb{Z}_p)$, для яких операція множення є комутативною:

$$\Gamma_1 = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}, t, a \in \mathbb{Z}_p, t \neq 0, a \neq 0 \right\},$$

$$\Gamma_2 = \left\{ t \cdot \begin{pmatrix} 1 & 0 \\ a & ak + 1 \end{pmatrix}, t, a, k \in \mathbb{Z}_p, t \neq 0, ak + 1 \neq 0 \right\}, \quad k - \text{фіксоване};$$

$$\Gamma_3 = \left\{ t \cdot \begin{pmatrix} 1 & a \\ 0 & ak+1 \end{pmatrix}, t, a, k \in \mathbb{Z}_p, t \neq 0, ak+1 \neq 0 \right\}, k - \text{фіксоване};$$

$$\Gamma_4 = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & 0 \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}, a \text{ і } b - \text{фіксовані};$$

$$\Gamma_5 = \left\{ t \cdot \begin{pmatrix} 0 & 1 \\ b & a \end{pmatrix}, t, a, b \in \mathbb{Z}_p, t \neq 0, b \neq 0 \right\}, a \text{ і } b - \text{фіксовані};$$

$$\Gamma_6 = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, t, a, b, k \in \mathbb{Z}_p, t \neq 0, b \neq 0, a(a+k)-b \neq 0 \right\}, b \text{ і } k -$$

фіксовані.

Твердження доведено. ■

Потужності сімейств $\Gamma_1 - \Gamma_3$ дорівнюють $(p-1)^2$, а сімейств $\Gamma_4 - \Gamma_5$ — $p-1$. Потужність же сімейства Γ_6 дорівнює $(p-1)(p-l)$, де $l = \{0; 1; 2\}$ — кількість цілих коренів рівняння $a^2 + ka - b = 0 \pmod{p}$ відносно змінної a . Значення l визначається параметрами b і k .

3.3. Комутативне сімейство матриць 2×2 з одиницею

Розглянемо сімейство матриць множини Γ_6 , доповненої одиничною матрицею, а також випадком, коли $b=0$, оскільки він не впливає на комутативність матриць множини Γ_6 . Позначимо отримане сімейство через

$$CGL_{b,k}(2, \mathbb{Z}_p) = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, t, s, a, b, k \in \mathbb{Z}_p, t, s \neq 0, a(a+k)-b \neq 0 \right\}$$

Твердження 2. Сімейство матриць $CGL_{b,k}(2, \mathbb{Z}_p)$ є комутативною (абелевою) групою за множенням.

Доведення.

Доведемо виконання аксіом групи для $CGL_{b,k}(2, \mathbb{Z}_p)$ і покажемо, що операція множення у $CGL_{b,k}(2, \mathbb{Z}_p)$ є комутативною.

1. У $CGL_{b,k}(2, \mathbb{Z}_p)$ існує єдиний нейтральний (одичний) елемент:

$$E = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

2. Операція множення елементів у $CGL_{b,k}(2, \mathbb{Z}_p)$ є асоціативною, оскільки це є загальною властивістю матриць.

3. Для кожної матриці $A \in CGL_{b,k}(2, \mathbb{Z}_p)$ існує обернена матриця $A^{-1} \in CGL_{b,k}(2, \mathbb{Z}_p)$: $A \cdot A^{-1} = A^{-1} \cdot A = E$.

Так, якщо $A = s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, то $A^{-1} = \left[s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right]^{-1} = s^{-1} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. У \mathbb{Z}_p для

кожного $s \in \mathbb{Z}_p$ існує обернений елемент s^{-1} : $s \cdot s^{-1} = s^{-1} \cdot s = e = 1$, причому єдиний. Надалі в доведенні цього твердження, не обмежуючи загальності, множниками s і t будемо нехтувати.

Нехай $A = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}$. Тоді $A^{-1} = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}^{-1} = \frac{1}{a(a+k)-b} \begin{pmatrix} a+k & -1 \\ -b & a \end{pmatrix}$.

Покладемо $t' = \frac{-1}{a(a+k)-b} \neq 0$ і $a' = -a-k$. Тоді

$$A^{-1} = t' \cdot \begin{pmatrix} a' & 1 \\ b & a'+k \end{pmatrix} \in CGL_{b,k}(2, \mathbb{Z}_p).$$

Крім того, звідси також слідує, що $\begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix} \begin{pmatrix} c & 1 \\ b & c+k \end{pmatrix}^{-1} = E$ тоді і

тільки тоді, коли $a=c$.

4. $CGL_{b,k}(2, \mathbb{Z}_p)$ є замкнутою відносно операції множення.

Нехай $A, B \in CGL_{b,k}(2, \mathbb{Z}_p)$. Якщо $A=E$ або $B=E$, ця властивість є очевидною.

Розглянемо ситуацію, коли $A = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}$ і $B = \begin{pmatrix} x & 1 \\ b & x+k \end{pmatrix}$ для довільних

$a, x \in \mathbb{Z}_p$.

Добуток $A \cdot B = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix} \begin{pmatrix} x & 1 \\ b & x+k \end{pmatrix} = \begin{pmatrix} ax+b & a+x+k \\ b(a+x+k) & b+(a+k)(x+k) \end{pmatrix}.$

Якщо $a+x+k=0$, то $A \cdot B = \begin{pmatrix} ax+b & 0 \\ 0 & ax+b \end{pmatrix} = (ax+b) \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$

Оскільки $|A| \neq 0$ і $|B| \neq 0$, то $|A \cdot B| = |A| \cdot |B| \neq 0$ і, відповідно, $ax+b \neq 0$. Звідси слідує, що $A \cdot B \in CGL_{b,k}(2, \mathbb{Z}_p)$.

Якщо $a+x+k \neq 0$, то $A \cdot B = \frac{1}{a+x+k} \begin{pmatrix} \frac{ax+b}{a+x+k} & 1 \\ b & k + \frac{ax+b}{a+x+k} \end{pmatrix}.$

Покладемо $t = \frac{1}{a+x+k} \neq 0$ і $y = \frac{ax+b}{a+x+k}$. Тоді

$$A \cdot B = t \begin{pmatrix} y & 1 \\ b & y+k \end{pmatrix} \in CGL_{b,k}(2, \mathbb{Z}_p).$$

5. Група $CGL_{b,k}(2, \mathbb{Z}_p)$ є абелевою.

Якщо $A = E$, то $A \cdot B = E \cdot B = B = B \cdot E = B \cdot A$.

Нехай $A = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}$ і $B = \begin{pmatrix} x & 1 \\ b & x+k \end{pmatrix}$. Тоді

$$A \cdot B = \begin{pmatrix} ax+b & a+x+k \\ b(a+x+k) & b+(a+k)(x+k) \end{pmatrix}, \text{ а}$$

$$B \cdot A = \begin{pmatrix} ax+b & a+x+k \\ b(a+x+k) & b+(x+k)(a+k) \end{pmatrix}, \text{ звідки } A \cdot B = B \cdot A.$$

Твердження доведено. ■

Для піднесення квадратної матриці $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ до степеня будемо

використовувати наступний вираз з [168]:

$$A^n = \begin{pmatrix} u_{n+1} - du_n & bu_n \\ cu_n & u_{n+1} - au_n \end{pmatrix}, \quad (3.1)$$

де $u_{n+1} = (a + d)u_n - |A|u_{n-1} = \text{tr}(A)u_n - |A|u_{n-1}$, $\text{tr}(A)$ – слід матриці A [169];

$$u_0 = 0, \quad u_1 = 1.$$

Для елемента групи $A = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix} \in CGL_{b,k}(2, \mathbb{Z}_p)$ маємо: $\text{tr}(A) = 2a + k$,
 $|A| = a(a+k) - b \neq 0$. Тоді $u_{n+1} = (2a + k)u_n - (a(a+k) - b)u_{n-1}$, а
 $A^n = \begin{pmatrix} u_{n+1} - (a+k)u_n & u_n \\ bu_n & u_{n+1} - au_n \end{pmatrix}$.

Зауважимо, що $|A^n| = |A|^n \neq 0$.

Оскільки $CGL_{b,k}(2, \mathbb{Z}_p)$ є комутативною групою за множенням, $A^n \in CGL_{b,k}(2, \mathbb{Z}_p)$. Відповідно до (3.1):

1) якщо $u_n = 0$, то $A^n = \begin{pmatrix} u_{n+1} & 0 \\ 0 & u_{n+1} \end{pmatrix} = u_{n+1} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \in CGL_{b,k}(2, \mathbb{Z}_p)$;

2) якщо $u_n \neq 0$, то $A^n = u_n \begin{pmatrix} \frac{u_{n+1}}{u_n} - a - k & 1 \\ b & \frac{u_{n+1}}{u_n} - a \end{pmatrix} \in CGL_{b,k}(2, \mathbb{Z}_p)$.

Твердження 3. Порядок групи $CGL_{b,k}(2, \mathbb{Z}_p)$ для $D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p$ дорівнює $p^2 - 1$.

Доведення.

Нагадаємо,

що

$$CGL_{b,k}(2, \mathbb{Z}_p) = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, t, s, a, b, k \in \mathbb{Z}_p, t, s \neq 0, a(a+k) - b \neq 0 \right\}$$

.

Оскільки значення b і k є фіксованими для групи, змінними є $t, a, s \in \mathbb{Z}_p$, $t, s \neq 0$. Тоді кількість різних значень матриць виду $t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}$ дорівнює

кількості різних можливих пар $\{t, a\}$ з заданими обмеженнями. Значення t може приймати $p-1$ різних значень з \mathbb{Z}_p ($t \neq 0$). Значення a обмежене умовою $a(a+k)-b \neq 0$. Для $D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p$ це рівняння не має цілих коренів відносно змінної a , тому вона може набувати p різних значень з \mathbb{Z}_p .

Кількість різних значень матриць виду $s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ дорівнює числу $p-1$ різних можливих значень $s \in \mathbb{Z}_p$, $s \neq 0$.

Таким чином, порядок групи $CGL_{b,k}(2, \mathbb{Z}_p)$ дорівнює $(p-1)p + p - 1 = p^2 - 1$.

Твердження доведено. ■

Таким чином, $CGL_{b,k}(2, \mathbb{Z}_p)$ для $D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p$ є мультиплікативною абелевою групою порядку $p^2 - 1$.

Зауваження 1 [170]. Кількість ненульових значень $D \in \mathbb{Z}_p : D = u^2 \in \mathbb{Z}_p$ і кількість значень $D \in \mathbb{Z}_p : D \neq u^2 \in \mathbb{Z}_p$ за простого $p \geq 3$ однакові й дорівнюють $\frac{p-1}{2}$.

Надалі прийmemo:

$$CGL_{b,k}(2, \mathbb{Z}_p) = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{matrix} t, s, a, b, k \in \mathbb{Z}_p, t, s \neq 0, \\ D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p \end{matrix} \right\}.$$

3.4. Діагоналізація матриць групи $CGL_{b,k}(2, \mathbb{Z}_p)$

Зазначимо, що відповідно до [167] перестановочні матриці простої структури можна одночасно, тобто одним і тим же перетворенням подібності, привести до діагонального виду.

Під матрицями простої структури розуміють матриці, які мають n лінійно незалежних власних векторів [167]. Оскільки власні вектори, які відповідають попарно різним характеристичним числам, завжди лінійно незалежні, для того, щоб матриця мала просту структуру, достатньо, щоб усі корені характеристичного рівняння були різні [167], [171].

Характеристичний поліном матриці $A = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix} \in CGL_{b,k}(2, \mathbb{Z}_p)$

дорівнює $|A - \lambda E| = \begin{vmatrix} a - \lambda & 1 \\ b & a + k - \lambda \end{vmatrix} = \lambda^2 - (2a + k)\lambda + a(a + k) - b$, де E – одинична матриця розмірності $n = 2$.

Дискримінант характеристичного рівняння $D = (2a + k)^2 - 4(a^2 + ak - b) = k^2 + 4b$. У випадку, коли значення D не є квадратичним лишком у простому полі лишків \mathbb{Z}_p ($D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p$), характеристичний поліном не має коренів у \mathbb{Z}_p . Оскільки степінь рівняння $n = 2$, поліном незвідний над полем \mathbb{Z}_p .

Розглянемо незвідний поліном $f(x) = x^2 - D \in \mathbb{Z}_p[x]$. Просте алгебраїчне розширення степеня 2 поля \mathbb{Z}_p задамо як $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$ [172], де $D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p$.

Поле Галуа F_{p^2} має характеристику p і степінь 2 [173].

Зауваження 2. Поле F_{p^2} є поле розкладання для характеристичних поліномів матриць із групи $CGL_{b,k}(2, \mathbb{Z}_p)$. Власні значення матриці

$tA = t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}$ над полем $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$:

$$\lambda_{1,2}(a, t) = \frac{t}{2} (2a + k \pm \sqrt{D}), \quad t \neq 0. \quad (1)$$

Дійсно, для матриці tA , $t \neq 0$, характеристичне рівняння $|tA - \lambda E| = t^2 \left| A - \frac{\lambda}{t} E \right| = 0$, звідки $\lambda_{1,2}(a, t) = t \cdot \lambda_{1,2}(a)$, де $\lambda_{1,2}(a)$ – власні

значення матриці $A = \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}$ над полем $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$.

Якщо $\lambda(a, t) = \frac{t}{2}(2a + k \pm \sqrt{D})$ є одним з коренів незвідного в \mathbb{Z}_p характеристичного рівняння, де $\lambda \in F_{p^2}$, то в силу теореми 2.14 з [173] інший корінь рівняння дорівнює $\lambda^p(a, t) = \frac{t}{2}(2a + k \mp \sqrt{D})$.

Для матриці sE власні значення $\lambda_{1,2}(s) = s \neq 0$.

Лема 1.3.19 з [171] засвідчує те, що сімейство діагоналізованих матриць є комутативним сімейством тоді і тільки тоді, коли воно одночасно діагоналізоване. На основі цієї леми сформулюємо наступне зауваження.

Зауваження 3. Комутативне сімейство матриць $CGL_{b,k}(2, \mathbb{Z}_p)$ над полем $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$ одночасно діагоналізоване, тобто існує матриця

$C = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$ з елементами з F_{p^2} , така, що для кожної матриці

$A \in CGL_{b,k}(2, \mathbb{Z}_p)$ добуток $C^{-1} \cdot A \cdot C$ є діагональною матрицею:

$$C^{-1} \cdot A \cdot C = \begin{pmatrix} \lambda_1(a, t) & 0 \\ 0 & \lambda_2(a, t) \end{pmatrix}.$$

Знайдемо таку матрицю C .

Оскільки $C^{-1} \cdot A \cdot C = \begin{pmatrix} \lambda_1(a, t) & 0 \\ 0 & \lambda_2(a, t) \end{pmatrix}$, то

$$t \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix} \cdot \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} \begin{pmatrix} \lambda_1(a, t) & 0 \\ 0 & \lambda_2(a, t) \end{pmatrix}. \text{ Виконавши множення}$$

матриць і їх порівняння, отримаємо:

$$\begin{cases} t(ac_{11} + c_{21}) = c_{11}\lambda_1(a, t), \\ t(bc_{11} + (a+k)c_{21}) = c_{21}\lambda_1(a, t), \\ t(ac_{12} + c_{22}) = c_{12}\lambda_2(a, t), \\ t(bc_{12} + (a+k)c_{22}) = c_{22}\lambda_2(a, t). \end{cases} \quad (3.2)$$

Враховуючи $\lambda_{1,2}(a, t) = t \cdot \lambda_{1,2}(a)$, перепишемо останню систему рівнянь:

$$\begin{cases} ac_{11} + c_{21} = c_{11}\lambda_1(a), \\ bc_{11} + (a+k)c_{21} = c_{21}\lambda_1(a), \\ ac_{12} + c_{22} = c_{12}\lambda_2(a), \\ bc_{12} + (a+k)c_{22} = c_{22}\lambda_2(a); \end{cases} \Rightarrow \begin{cases} (a - \lambda_1(a))c_{11} + c_{21} = 0, \\ bc_{11} + (a+k - \lambda_1(a))c_{21} = 0, \\ (a - \lambda_2(a))c_{12} + c_{22} = 0, \\ bc_{12} + (a+k - \lambda_2(a))c_{22} = 0. \end{cases} \quad (3.3)$$

Прийmemo $\lambda_1(a) = \frac{2a+k+\sqrt{D}}{2}$. Тоді $\lambda_2(a) = \frac{2a+k-\sqrt{D}}{2}$.

З перших двох рівнянь системи маємо:

$$\begin{cases} c_{21} = \frac{k+\sqrt{D}}{2}c_{11}, \\ bc_{11} + \left(\frac{k-\sqrt{D}}{2}\right)c_{21} = 0. \end{cases} \quad (3.4)$$

Прийmemo $c_{11} = 1$ у (3.4). Тоді $c_{21} = \frac{k+\sqrt{D}}{2}$ і перший власний вектор матриці C дорівнює:

$$\bar{e}_1 = \begin{pmatrix} 1 \\ \frac{k+\sqrt{D}}{2} \end{pmatrix}. \quad (3.5)$$

Аналогічно, з інших двох рівнянь системи знаходимо другий власний вектор матриці C :

$$\bar{e}_2 = \begin{pmatrix} 1 \\ \frac{k-\sqrt{D}}{2} \end{pmatrix}. \quad (3.6)$$

Тоді $C = \begin{pmatrix} 1 & 1 \\ \frac{k + \sqrt{D}}{2} & \frac{k - \sqrt{D}}{2} \end{pmatrix}$. Відповідно матриця C не залежить від

значень a і t та є спільною для $CGL_{b,k}(2, \mathbb{Z}_p)$.

Виконаємо перевірку.

$$\begin{aligned} C^{-1} \cdot A \cdot C &= \frac{1}{|C|} \begin{pmatrix} \frac{k - \sqrt{D}}{2} & -1 \\ -\frac{k + \sqrt{D}}{2} & 1 \end{pmatrix} t \begin{pmatrix} a & 1 \\ b & a + k \end{pmatrix} \begin{pmatrix} 1 & 1 \\ \frac{k + \sqrt{D}}{2} & \frac{k - \sqrt{D}}{2} \end{pmatrix} = \\ &= t \begin{pmatrix} \frac{2a + k + \sqrt{D}}{2} & 0 \\ 0 & \frac{2a + k - \sqrt{D}}{2} \end{pmatrix} = \begin{pmatrix} \lambda_1(a, t) & 0 \\ 0 & \lambda_2(a, t) \end{pmatrix}. \end{aligned}$$

Розглянемо множину D_λ невідроджених діагональних матриць над полем $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$:

$$D_\lambda = \left\{ \begin{pmatrix} \lambda & 0 \\ 0 & \lambda^p \end{pmatrix}, \lambda \in F_{p^2} \right\}. \quad (3.7)$$

Зауваження 4. Відображення $g(A) = C^{-1} \cdot A \cdot C$ задає взаємнооднозначну відповідність (бієкцію) між матрицями $A \in CGL_{b,k}(2, \mathbb{Z}_p)$ та діагональними матрицями з D_λ , тобто $g : CGL_{b,k}(2, \mathbb{Z}_p) \leftrightarrow D_\lambda$.

Доведення.

Якщо задано матриці $A_1, A_2 \in CGL_{b,k}(2, \mathbb{Z}_p)$, $A_1 \neq A_2$, а λ_1, λ_1^p і λ_2, λ_2^p – власні значення матриць A_1 і A_2 відповідно, то

$$C^{-1} \cdot \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_1^p \end{pmatrix} \cdot C \neq C^{-1} \cdot \begin{pmatrix} \lambda_2 & 0 \\ 0 & \lambda_2^p \end{pmatrix} \cdot C \Leftrightarrow \lambda_1 \neq \lambda_2.$$

Кількість різних матриць множини D_λ дорівнює $p^2 - 1$, що відповідає порядку мультиплікативної абелевої групи $CGL_{b,k}(2, \mathbb{Z}_p)$. Це означає, що

відображення $g = g(A)$ встановлює взаємнооднозначну відповідність між $CGL_{b,k}(2, \mathbb{Z}_p)$ і D_λ . ■

$$\text{Позначимо через } F_{b,k} = \left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, D = k^2 + 4b \neq u^2 \in \mathbb{Z}_p, \right.$$

сімейство матриць, де p – просте, b, k – фіксовані в \mathbb{Z}_p .

Твердження 4. Сімейство матриць $F_{b,k}$ утворює поле Галуа порядку p^2 зі звичайними операціями множення та додавання матриць.

Доведення.

$$\text{Очевидно, що } F_{b,k} = CGL_{b,k}(2, \mathbb{Z}_p) \cup \Theta, \text{ де } \Theta = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Покажемо замкнутість операції додавання в множині $F_{b,k}$.

Відповідно до зауважень 3 і 4, для довільних матриць A_1 і A_2 з $F_{b,k}$ існує одна й та ж матриця C така, що:

$$\begin{cases} A_1 = C \cdot \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_1^p \end{pmatrix} \cdot C^{-1}, \\ A_2 = C \cdot \begin{pmatrix} \lambda_2 & 0 \\ 0 & \lambda_2^p \end{pmatrix} \cdot C^{-1}; \end{cases} \Rightarrow A_1 + A_2 = C \cdot \begin{pmatrix} \lambda_1 + \lambda_2 & 0 \\ 0 & \lambda_1^p + \lambda_2^p \end{pmatrix} \cdot C^{-1}. \quad (3.8)$$

Поле Галуа $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$ має характеристику p . Тому в силу пропозиції 7.1.4 із [172] виконується рівність $\lambda_1^p + \lambda_2^p = (\lambda_1 + \lambda_2)^p$. Тоді для

$$\lambda_3 = \lambda_1 + \lambda_2: \quad A_1 + A_2 = C \cdot \begin{pmatrix} \lambda_3 & 0 \\ 0 & \lambda_3^p \end{pmatrix} \cdot C^{-1} \quad \text{або} \quad C^{-1} \cdot (A_1 + A_2) \cdot C = \begin{pmatrix} \lambda_3 & 0 \\ 0 & \lambda_3^p \end{pmatrix} \in D_\lambda.$$

Очевидно, що $C^{-1} \cdot \Theta \cdot C = \Theta$. Згідно з зауваженням 4, існує єдина матриця

$$A_3 \in CGL_{b,k}(2, \mathbb{Z}_p), \quad \text{що} \quad C^{-1} \cdot A_3 \cdot C = \begin{pmatrix} \lambda_3 & 0 \\ 0 & \lambda_3^p \end{pmatrix}, \quad \lambda_3 \in F_{p^2}. \quad \text{Відповідно,}$$

$$A_1 + A_2 = A_3 \in F_{b,k}.$$

Таким чином, сімейство матриць $F_{b,k}$ може бути представлене з допомогою фіксованої матриці C у вигляді:

$$F_{b,k} = \left\{ C \cdot \begin{pmatrix} \lambda & 0 \\ 0 & \lambda^p \end{pmatrix} \cdot C^{-1}, \lambda \in F_{p^2} = \mathbb{Z}_p[\sqrt{D}] \right\}.$$

Легко бачити, що $F_{b,k}$ є алгебраїчним полем для звичайних операцій над матрицями, а його порядок дорівнює p^2 . ■

Наслідок 1. Мультиплікативна група $F_{b,k}^*$ скінченного поля $F_{b,k}$ циклічна, тобто група $CGL_{b,k}(2, \mathbb{Z}_p)$ – циклічна.

Наслідок 2. У полі $F_{b,k}$ кількість примітивних елементів дорівнює $\varphi(p^2 - 1)$, де $\varphi(m)$ – функція Ейлера від m .

3.5. Програмна модель формування ключів-перестановок через квадратну матрицю

Сумісна реалізація матричного представлення даних, зокрема, для узгодження криптографічних ключів, а також факторіального кодування інформації для її захищеного передавання каналами зв'язку призводить до необхідності трансформації отриманого ключа в вигляді матриці в ключ у вигляді перестановки. Враховуючи те, що довжини ключів і інформаційних блоків для протоколів на основі матричних перетворень і протоколів факторіального кодування в загальному випадку не співпадають, відображення матриць у перестановки не є бієктивним, тобто сюр'єктивним і ін'єктивним одночасно [146]. Відповідно, виникає потреба в формуванні та дослідженні алгоритмів відображення матриць у перестановки.

У цій частині роботи розглядається метод формування перестановок з квадратних матриць другого порядку.

Наводиться опис методу формування перестановки з використанням двох алгоритмів перетворення. З метою визначення найоптимальнішого

алгоритму перетворення матриці заданого порядку в перестановку наведено алгоритму роботи програмної моделі, що реалізує перетворення матриць.

3.5.1. Опис методу формування перестановки з квадратної матриці

За визначенням, наведеним в [174], квадратною матрицею є матриця, у якої кількість рядків дорівнює кількості стовпців: $i = j$. Тоді $A = (a_{ij})_{n \times n}$ є квадратною матрицею порядку n . Елементи матриці a_{ij} лежать у діапазоні $0 \leq a_{ij} \leq p-1$, де p – деяке ціле число; i, j – рядок і стовпець матриці відповідно. Запропоновано два алгоритми перетворення матриці A в перестановку:

- шляхом представлення матриці A в десятковій та факторіальній системах числення;
- шляхом використання допоміжних квадратних матриць.

Метод перетворення матриці A за першим алгоритмом має наступні етапи:

- 1) перетворення матриці A в десяткове значення A_{10} через обчислення:

$$A_{10} = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \cdot p^{(i-1)n+j-1} = a_{11} \cdot p^0 + a_{12} \cdot p^1 + \dots + a_{nn} \cdot p^{n^2-1}; \quad (3.9)$$

- 2) приведення десяткового еквіваленту A_{10} матриці A до діапазону $[0; M!-1]$:

$$A_{10}^{norm} = |A_{10}|_{M!}; \quad (3.10)$$

- 3) перетворення десяткового представлення числа A_{10}^{norm} з діапазону $[0; M!-1]$ у факторіальне число A_F та перестановку π довжини M :

$$A_{10}^{norm} \rightarrow A_F \rightarrow \pi. \quad (3.11)$$

Процес перетворення (3.11) є нескладним і не розглядається.

Другий метод перетворення матриці в перестановку полягає в наступному:

- 1) формування допоміжних матриць $H_k = (h_{ij}(k))_{n \times n}$, де
- $$h_{ij}(k) = \left| (k-1) \cdot n^2 + (i-1) \cdot n + j - 1 \right|_M, \text{ а } 1 \leq k \leq \left\lceil \frac{M}{n^2} \right\rceil.$$

Таким чином, коли значення M перевищує n^2 , допоміжних матриць H_k існує декілька. Наприклад, для $M = 8$, $n = 2$ формують дві допоміжні

$$\text{матриці: } H_1 = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}, H_2 = \begin{pmatrix} 4 & 5 \\ 6 & 7 \end{pmatrix}.$$

Для $M = 9$, $n = 2$ утворюють три допоміжні матриці:

$$H_1 = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}, H_2 = \begin{pmatrix} 4 & 5 \\ 6 & 7 \end{pmatrix}, H_3 = \begin{pmatrix} 8 & 0 \\ 1 & 2 \end{pmatrix};$$

- 2) почергове множення зліва матриці A на всі допоміжні матриці H_k :

$$B_k = A \cdot H_k = (b_{ij}(k))_{n \times n}; \quad (3.12)$$

- 3) формування факторіальних коефіцієнтів a_l , $0 \leq l \leq M - 1$ числа A_F шляхом обчислення:

$$a_l = \left| b_{ij}(k) \right|_{l+1}, \quad (3.13)$$

де значення i , j , k , що відповідають коефіцієнту a_l , обирають таким чином, щоб $l = h_{ij}(k)$;

- 4) перетворення факторіального числа A_F у перестановку π довжини M :

$$A_F \rightarrow \pi. \quad (3.14)$$

3.6. Програмна модель перетворення матриць другого порядку в перестановки та алгоритми роботи

Для описаних алгоритмів у розділі 3.5.1. необхідно виконати збір і аналіз статистичних даних щодо появи перестановок у результаті перебору всіх квадратних матриць заданого порядку. Постають задачі: формування всіх можливих варіацій матриці, збереження сформованих факторіальних чисел та обрахунку необхідних статистичних показників. Засобом для вирішення цих

задач обрано програмне середовище Matlab або його аналог у вільному доступі Octave, оскільки воно має необхідні готові інструменти для роботи з матрицями, обрахунку статистичних показників та можливість реалізації описаних перетворень шляхом програмування. З цією метою створено програмну модель перебору матриць, що реалізує два описаних алгоритми перетворення матриці в перестановку.

На етапі перетворення отриманого десяткового числа за формулою (3.14) в перестановку створено функцію перетворення *int2perm(number, base_permut)* у програмному середовищі Matlab. Функція приймає вхідними параметрами *number* – число в десятковому представленні, яке необхідно перетворити в перестановку. Параметр *base_permut* – базова перестановка у вигляді вектору (рядка). Блок-схему алгоритму функції, що реалізує перетворення десяткового числа в перестановку, представлено на (Рисунок 3.1)

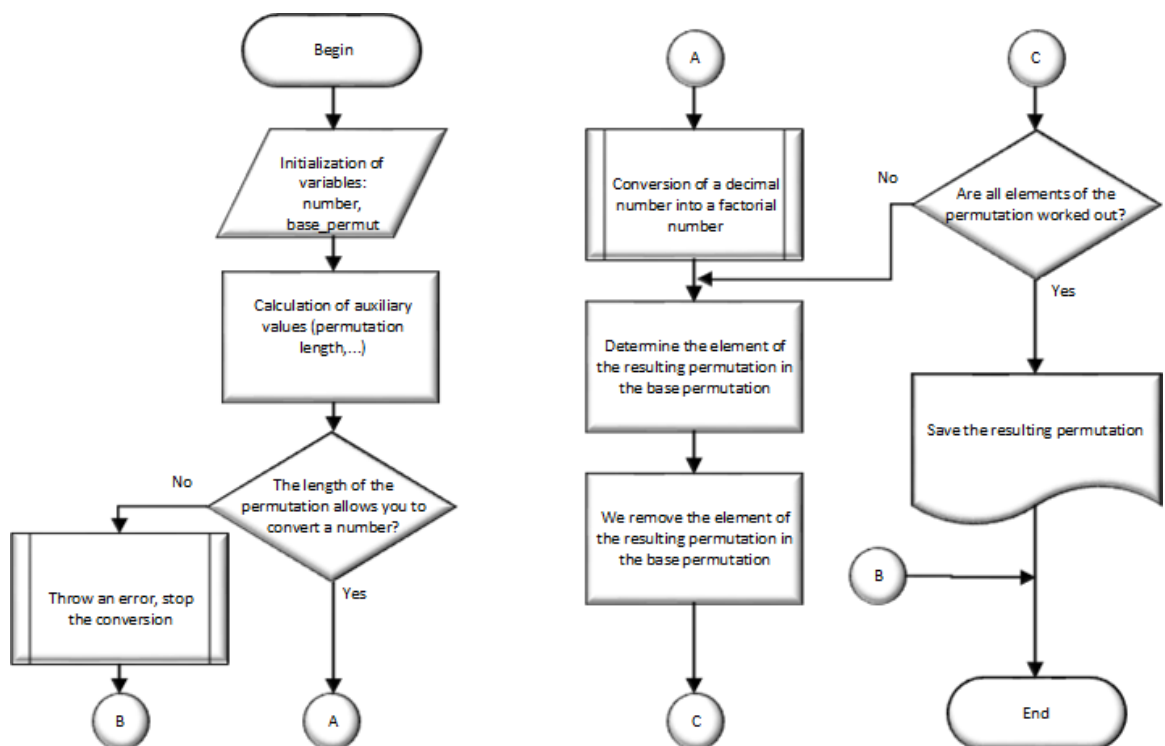


Рисунок 3.1 — Блок-схема алгоритму перетворення числа в перестановку (функція *int2perm*)

Для представлення числа в факторіальній і десятковій системах числення створено два додаткових методи: *int2fact()* – функція перетворення числа з десяткової в факторіальну систему числення, *fact2int()* – функція зворотного перетворення числа з факторіальної в десяткову систему числення.

Основні змінні, які використані під час побудови програмної моделі:

- *A* – квадратна матриця;
- *n* – порядок матриці *A*;
- *M* – довжина перестановки;
- *pMax* – максимальне значення елементів матриці *A* (змінна, що дозволяє реалізувати перевірку умови $a_{ij} \leq p-1$);
- *Adec* – поточне значення отриманого десяткового числа з матриці *A* за першим алгоритмом перетворення (змінна, що містить результат перетворення за формулою (3.9) та представлена в десятковій системі числення);
- *Bdec* – поточне значення отриманого числа з матриці *A* за другим алгоритмом перетворення (змінна, що містить результат перетворення за формулою (3.13) та приведена до десяткової системи числення);
- *AdecCollection* – список перетворених значень матриці за першим алгоритмом перетворення (змінна є одновимірним масивом (вектором), де кожен елемент є результатом поетапних перетворень матриці *A* за першим алгоритмом *Adec*);
- *BdecCollection* – список перетворених значень матриці за другим алгоритмом перетворення (змінна є одновимірним масивом (вектором), де кожен елемент є результатом поетапних перетворень матриці *A* за другим алгоритмом *Bdec*);
- *aCollection* – список сформованих значень матриці *A* (змінну представлено в вигляді тривимірного масиву. Два виміри визначають рядок і стовпець, третій вимір відповідає порядковому номеру кожної створеної матриці в діапазоні $[1; p^{n^2}]$).

3.6.1. Алгоритм роботи програмної моделі перебору матриць

Порядок дій програмної моделі полягає в наступному:

1. Введення параметрів: M , $pMax$, n .
2. Обрахунок значень: числа $M!$, кількості можливих варіантів матриці A , кількості допоміжних матриць для обробки за другим алгоритмом перетворення.
3. Перебір усіх варіантів матриці A за заданих умов, перетворення їх за першим та другим алгоритмами.
4. Збереження сформованих значень.

У результаті роботи програмної моделі отримуємо статистичні дані для подальшого оцінювання у вигляді розширення робочого простору **.mat*.

Блок схема описаного вище алгоритму має вигляд, наведений на (Рисунок 3.2).

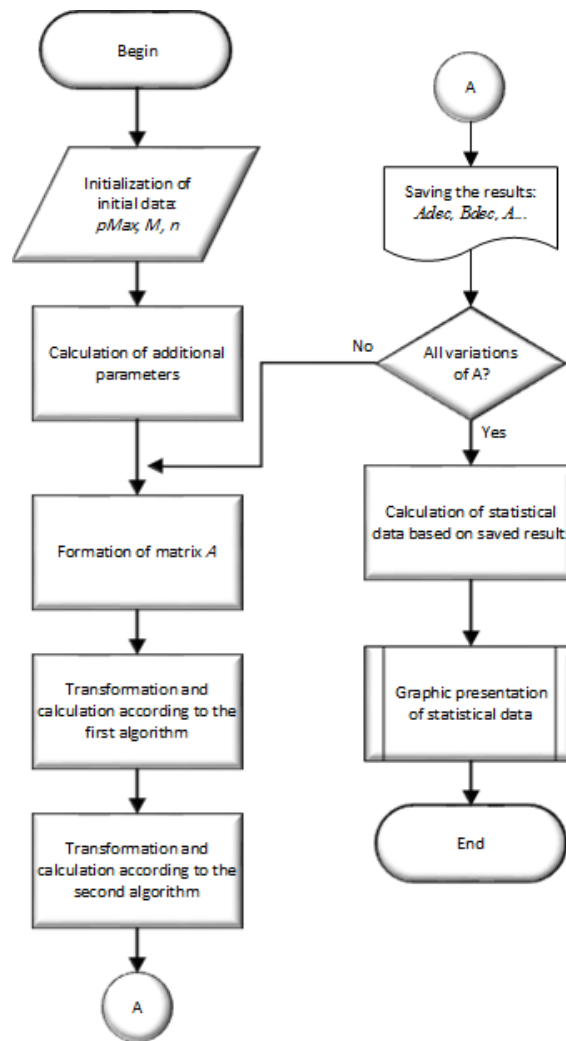


Рисунок 3.2 — Блок-схема алгоритму роботи програмної моделі

3.6.2. Особливості програмної моделі

Отримані результати проведених експериментів програмний продукт Matlab зберігає у вигляді розширення робочого простору **.mat*. У цьому форматі можна зберегти всі створені змінні під час роботи програми. Збереження робочого простору програми дозволяє виконати аналіз результатів моделювання й отримати доступ до масивів значень змінних окремо від основного алгоритму моделі. Тобто промодельовати експерименти та зберегти результати окремо для кожного із експериментів. У результаті після запуску програми моделювання у двох експериментах виконано збереження робочого простору для кожного моделювання. Отримано два файли, що містять змінні моделювання та їх значення.

Під час реалізації алгоритму використано вбудовані функції Matlab: *min* – мінімальне значення в числовій вибірці; *max* – максимальне значення в числовій вибірці; *range* – діапазон значень у числовій вибірці; *num* – обсяг вибірки; *mean* – середнє значення вибірки; *median* – медіана значень вибірки; *standart deviation* – середньоквадратичне відхилення.

Перелічені значення отримуються за допомогою функції Matlab *datastats()*. Функція *var()* дозволяє обрахувати дисперсію. Для відображення гістограм абсолютної частоти використано метод *bar(x,y,*)*, де *x* – вектор значень по осі *Ox*; *y* – вектор значень по осі *Oy*; *** – додаткові параметри налаштування (товщина ліній, колір та ін.). Для відображення графіків відносної частоти використано метод *plot(x,y,*)*.

3.6.3. Збір статистичних даних і їх валідація

Моделювання описаних алгоритмів перетворення матриці в перестановку виконано шляхом проведення двох експериментів з наступними вхідними параметрами:

- 1) $p = 17, M = 8, n = 2$;
- 2) $p = 23, M = 8, n = 2$.

Потужність множини матриці *A* дорівнює $17^4 = 83521$ для першого експерименту та $23^4 = 279841$ – для другого. Сформовані масиви даних, отримані в результаті виконання експериментів, є наборами номерів перестановок A_{10}^{norm} для першого алгоритму та отримані комбінації матриці *A*.

Для другого алгоритму перетворення збережений масив результатів містить номери перестановок отримані переведенням факторіального представлення у десяткове значень A_F . Фрагменти результатів для першого та другого експериментів наведено на (Рисунок 3.3) та (Рисунок 3.4).

	1	2	3
19	290		
20	291		
21	292		
22	293		
23	294		
24	295		
25	296		
26	297		
27	298		

	1	2	3
19	3602		
20	2260		
21	792		
22	4370		
23	2908		
24	1440		
25	98		
26	3676		
27	2208		

	1	2
val(:, :, 19) =	1	1
	0	0
val(:, :, 20) =		
	2	1
	0	0

Рисунок 3.3 — Сформовані та збережені результати перетворень за першим та другим алгоритмами для $p = 17, M = 8, n = 2$

	1	2
61	1072	
62	1073	
63	1074	
64	1075	
65	1076	
66	1077	

	1	2
61	76	
62	3648	
63	2186	
64	724	
65	4416	
66	2954	

Рисунок 3.4 — Сформовані та збережені результати перетворень за першим та другим алгоритмами для $p = 23, M = 8, n = 2$

Змінна *AdecCollection* містить результати перетворення за першим алгоритмом (значення A_{10}^{norm}). Змінна *BdecCollection* містить результати перетворення за другим алгоритмом (десятькове представлення номера перестановки отриманого A_F). Змінна *aCollection* містить значення всіх сформованих матриць A .

У результаті проведення першого експерименту з вхідними параметрами $p = 17, M = 8, n = 2$ отримано такі статистичні результати для перетворення за першим алгоритмом:

- кількість сформованих комбінацій матриці A , атрибут *num* у функції *datastat()*: $N = 83521$;

- максимальне значення числа, отриманого в результаті перетворення матриці A : $N_{\max} = M! - 1 = 40319$;
- мінімальне значення числа отриманого в результаті перетворення матриці A : $N_{\min} = 0$;
- середнє значення у сформованій вибірці: $mean = 19514$;
- медіана значень у сформованій вибірці: $median = 19439$;
- максимальна кількість повторів перестановок: $rep_{\max}^{A1} = 3$;
- мінімальна кількість повторів перестановок: $rep_{\min}^{A1} = 2$;
- значення A_{10}^{norm} з діапазону $[0; 2880]$ мають кількість повторів rep_{\max}^{A1} ;
- значення A_{10}^{norm} з діапазону $[2881; 40320)$ мають кількість повторів rep_{\min}^{A1} ;
- дисперсія: $D_{A1} = 142498178,841$;
- максимальне значення відносної частоти появи перестановки: $W_{\max}^{A1} = 3,5919 \cdot 10^{-5}$;
- мінімальне значення відносної частоти появи перестановки: $W_{\min}^{A1} = 2,3946 \cdot 10^{-5}$.

Гістограму розподілу абсолютних частот N_j , $0 \leq j \leq M! - 1$, появи перестановок з номером $x_j = j$ у лексикографічному порядку їх слідування за першим алгоритмом перетворення наведено на (Рисунок 3.5).

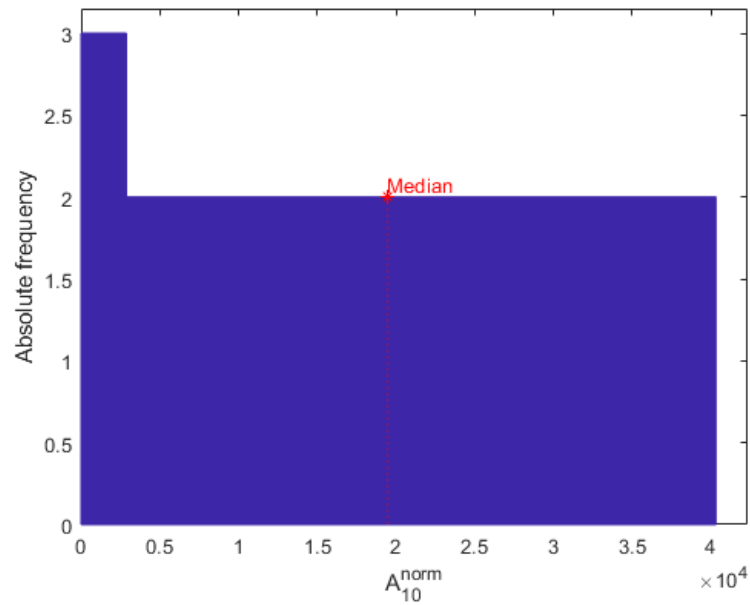


Рисунок 3.5 — Гістограма розподілу абсолютних частот появи перестановок за першим алгоритмом перетворення за результатами першого експерименту

Графік відносної частоти появи перестановок за першим алгоритмом перетворення наведено на (Рисунок 3.6). Вісь Ox містить сформовані значення порядкових номерів перестановок при перетворенні матриці A за формулою – A_{10}^{norm} , вісь Oy – відносна частота сформованого числа.

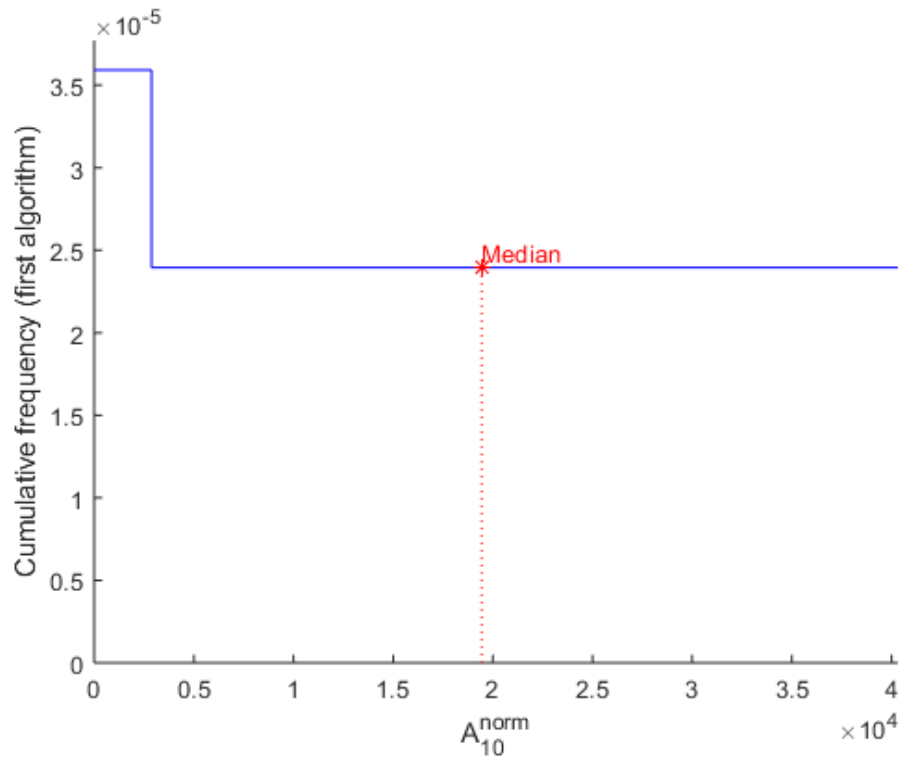


Рисунок 3.6 — Графік відносної частоти появи перестановок за першим алгоритмом перетворення за результатами першого експерименту

У результаті проведення експерименту з вхідними параметрами $p = 17$, $M = 8$, $n = 2$ отримано такі статистичні результати для перетворення за другим алгоритмом:

- кількість сформованих комбінацій матриці A , атрибут *num* у функції *datastat()*: $N = 83521$;
- максимальне значення числа, отриманого в результаті перетворення матриці A : $N_{\max} = 40198$;
- мінімальне значення числа, отриманого в результаті перетворення матриці A : $N_{\min} = 0$;
- середнє значення у сформованій вибірці: $mean = 19995$;
- медіана значень у сформованій вибірці, $median = 20012$;
- максимальна кількість повторів сформованих чисел (порядковий номер перестановки), $rep_{\max}^{B1} = 52$;

- мінімальна кількість повторів сформованих чисел (порядковий номер перестановки), $rep_{\min}^{B1} = 0$;
- кількість значень порядкових номерів перестановок, які мають максимальну кількість повторів $rep_{\max}^{B1} - 408$;
- кількість значень порядкових номерів перестановок, які мають мінімальну кількість повторів $rep_{\min}^{B1} - 38159$;
- дисперсія: $D_{B1} = 136109307,2089$;
- максимальне значення відносної частоти появи перестановки:
 $W_{\max}^{B1} = 62,2597 \cdot 10^{-5}$;
- мінімальне значення відносної частоти появи перестановки:
 $W_{\min}^{B1} = 35,9191 \cdot 10^{-5}$.

Гістограму розподілу абсолютних частот N_j , $0 \leq j \leq M!-1$, появи перестановок з номером $x_j = j$ у лексикографічному порядку їх слідування за першим алгоритмом перетворення наведено на (Рисунок 3.7).

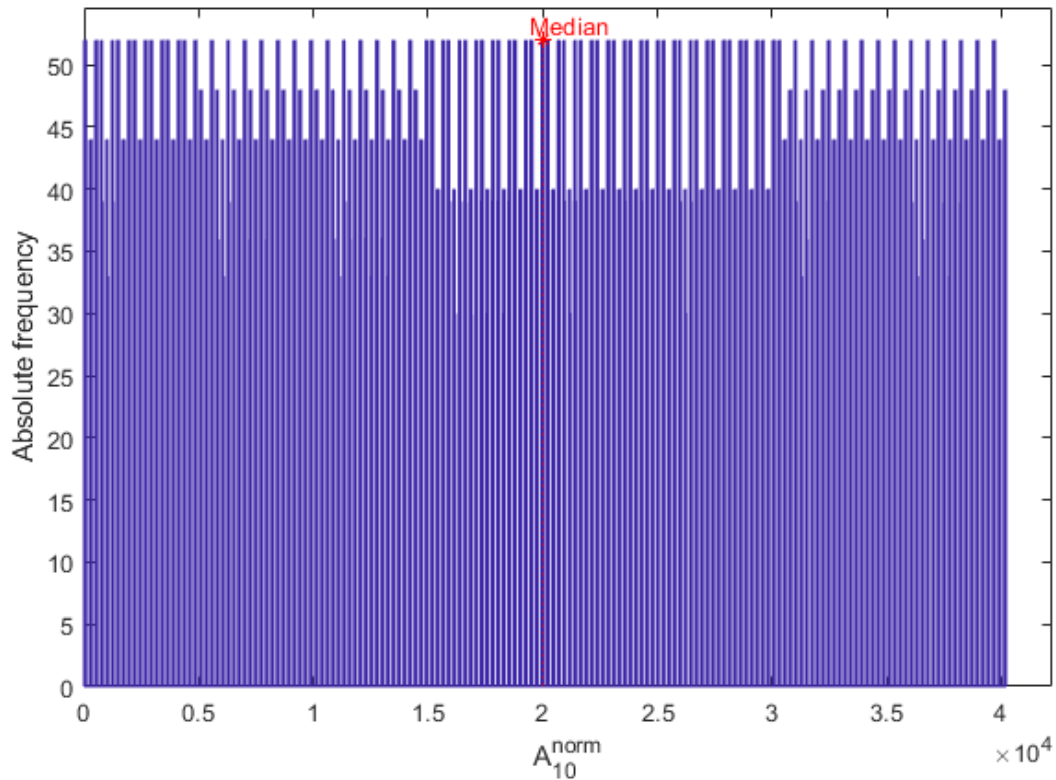


Рисунок 3.7 — Гістограма розподілу абсолютних частот появи перестановок за другим алгоритмом перетворення, за результатами першого експерименту

Графік відносної частоти появи перестановок за першим алгоритмом перетворення наведено на (Рисунок 3.8). Вісь Ox містить сформовані значення порядкових номерів перестановок при перетворенні матриці A за формулою – A_{10}^{norm} , вісь Oy – відносна частота сформованого числа.

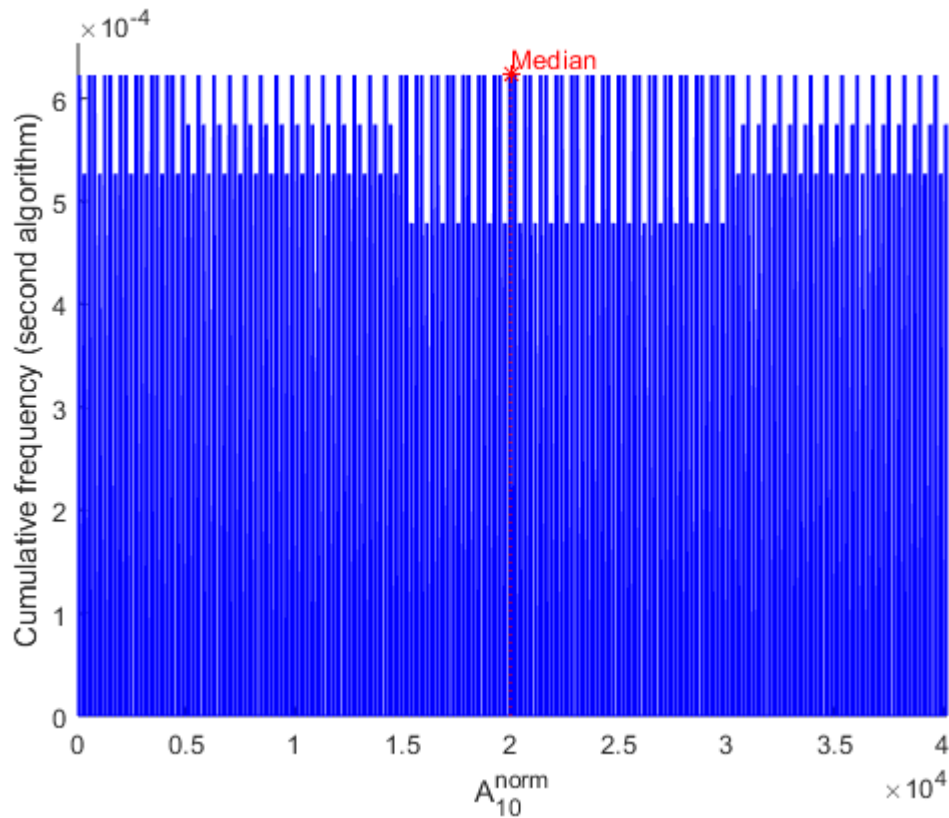


Рисунок 3.8 — Графік відносної частоти появи перестановок за другим алгоритмом перетворення, за результатами першого експерименту

У результаті проведення другого експерименту з вхідними параметрами $p = 23$, $M = 8$, $n = 2$ отримано такі статистичні результати для перетворення за першим алгоритмом:

- кількість сформованих комбінацій матриці A , атрибут *num* у функції *datastat()*: $N = 279841$;
- максимальне значення числа, отриманого в результаті перетворення матриці A : $N_{\max} = M! - 1 = 40319$;
- мінімальне значення числа отриманого в результаті перетворення матриці A : $N_{\min} = 0$;
- середнє значення у сформованій вибірці: $mean = 19996,957$;
- медіана значень у сформованій вибірці: $median = 19988$;
- максимальна кількість повторів перестановок: $rep_{\max}^{A1} = 7$;
- мінімальна кількість повторів перестановок: $rep_{\min}^{A1} = 6$;

- значення A_{10}^{norm} з діапазону $[0;37920]$ мають кількість повторів rep_{max}^{A1} ;
- значення A_{10}^{norm} з діапазону $[37921;40320)$ мають кількість повторів rep_{min}^{A1} ;
- дисперсія: $D_{A1} = 133524637,447$;
- максимальне значення відносної частоти появи перестановки: $W_{max}^{A1} = 2,5014 \cdot 10^{-5}$;
- мінімальне значення відносної частоти появи перестановки: $W_{min}^{A1} = 2,1441 \cdot 10^{-5}$.

Гістограму розподілу абсолютних частот N_j , $0 \leq j \leq M!-1$, появи перестановок з номером $x_j = j$ у лексикографічному порядку їх слідування за першим алгоритмом перетворення наведено на (Рисунок 3.9).

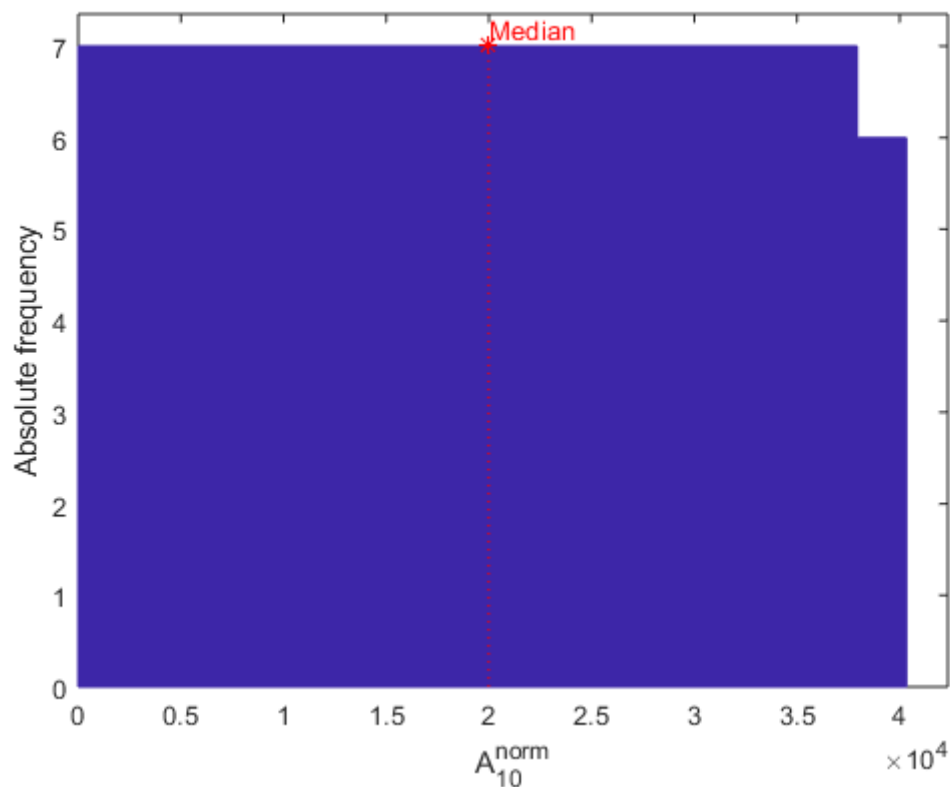


Рисунок 3.9 — Гістограма розподілу абсолютних частот появи перестановок за першим алгоритмом перетворення, за результатами другого експерименту

Графік відносної частоти появи перестановок за першим алгоритмом перетворення наведено на (Рисунок 3.10). Вісь Ox містить сформовані значення порядкових номерів перестановок при перетворенні матриці A за формулою – A_{10}^{norm} , вісь Oy – відносна частота сформованого числа.

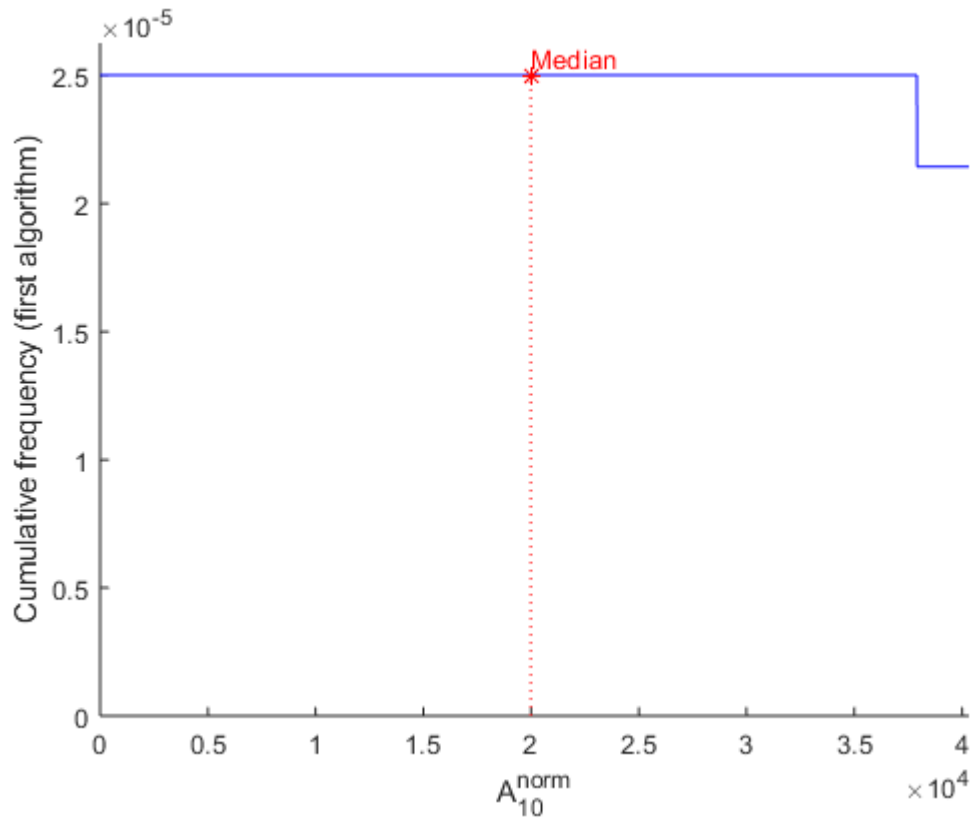


Рисунок 3.10 — Графік відносної частоти появи перестановок за першим алгоритмом перетворення за результатами другого експерименту

У результаті проведення експерименту з вхідними параметрами $p = 23$, $M = 8$, $n = 2$ отримано такі статистичні результати для перетворення за другим алгоритмом:

- кількість сформованих комбінацій матриці A , атрибут *num* у функції *datastat()*: $N = 279841$;
- максимальне значення числа, отриманого в результаті перетворення матриці A : $N_{\max} = 40198$;

- мінімальне значення числа, отриманого в результаті перетворення матриці A : $N_{\min} = 0$;
- середнє значення у сформованій вибірці: $mean = 20072,2193$;
- медіана значень у сформованій вибірці, $median = 20162$;
- максимальна кількість повторів сформованих чисел (порядковий номер перестановки), $rep_{\max}^{B1} = 230$;
- мінімальна кількість повторів сформованих чисел (порядковий номер перестановки), $rep_{\min}^{B1} = 0$;
- кількість значень порядкових номерів перестановок, які мають максимальну кількість повторів $rep_{\max}^{B1} - 64$;
- кількість значень порядкових номерів перестановок, які мають максимальну кількість повторів $rep_{\min}^{B1} - 37679$;
- дисперсія: $D_{B1} = 1335421877,799$;
- максимальне значення відносної частоти появи перестановки: $W_{\max}^{B1} = 82,1895 \cdot 10^{-5}$;
- мінімальне значення відносної частоти появи перестановки: $W_{\min}^{B1} = 28,5877 \cdot 10^{-5}$.

Гістограму розподілу абсолютних частот N_j , $0 \leq j \leq M! - 1$, появи перестановок з номером $x_j = j$ у лексикографічному порядку їх слідування за першим алгоритмом перетворення наведено на (Рисунок 3.11).

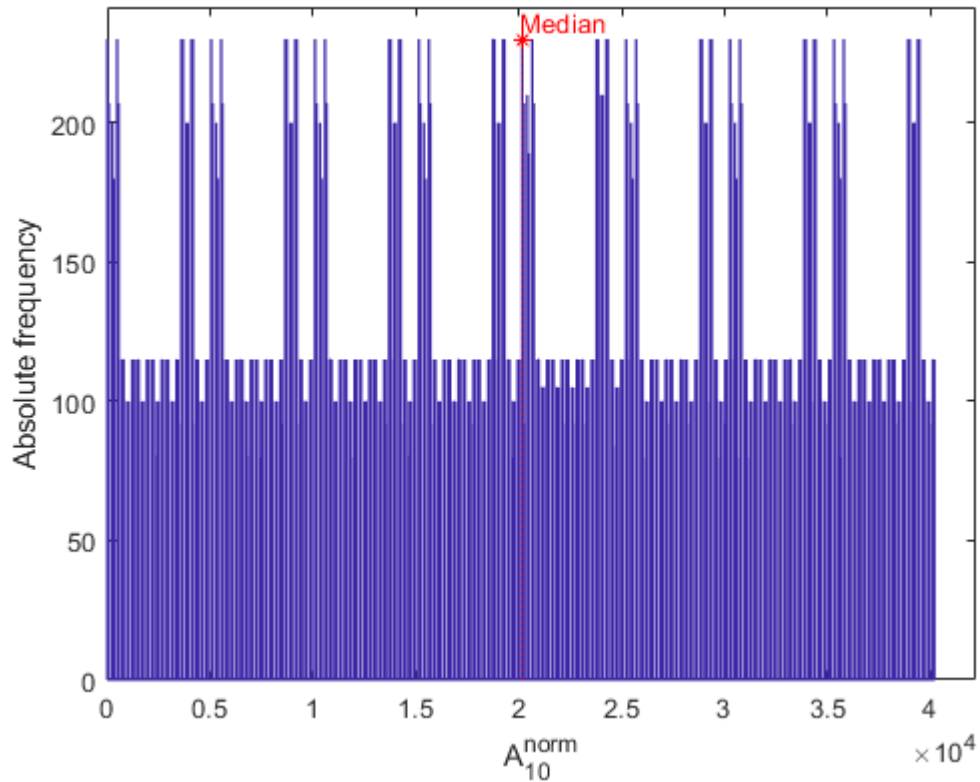


Рисунок 3.11 — Гістограма розподілу абсолютних частот появи перестановок за другим алгоритмом перетворення за результатами другого експерименту

Графік відносної частоти появи перестановок за другим алгоритмом перетворення наведено на (Рисунок 3.12). Вісь Ox містить сформовані значення порядкових номерів перестановок при перетворенні матриці A за формулою – A_{10}^{norm} , вісь Oy – відносна частота сформованого числа.

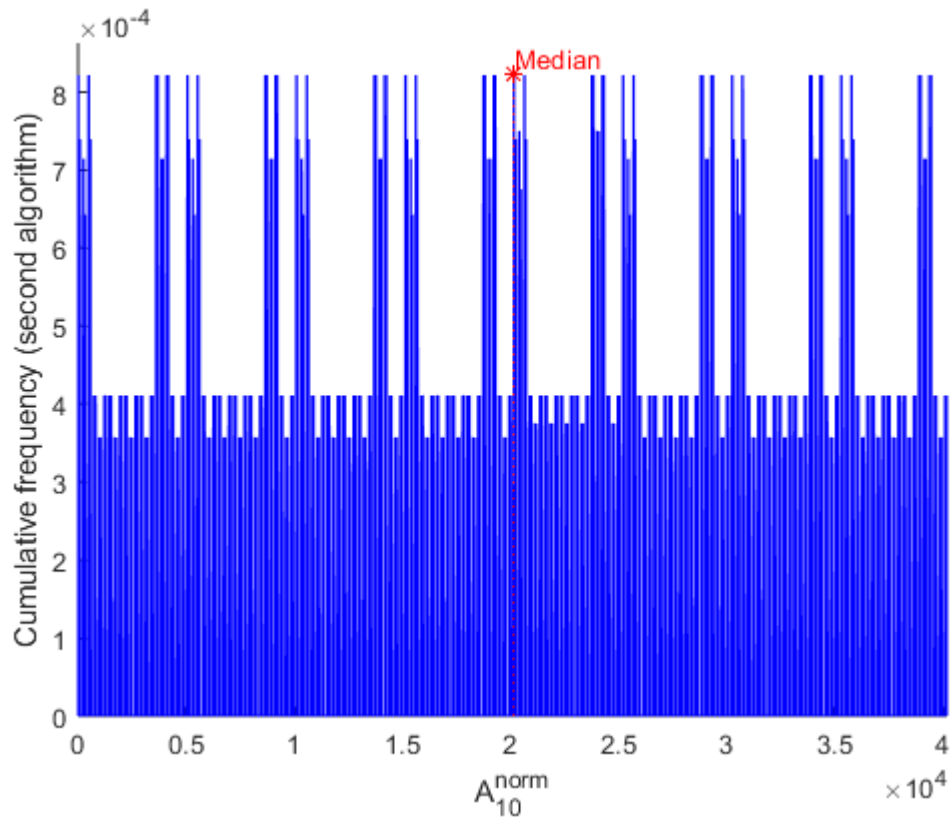


Рисунок 3.12 — Графік відносної частоти появи перестановок за другим алгоритмом перетворення за результатами другого експерименту

Статистичні результати першого та другого експериментів в таблиці нижче.

Таблиця 3.1 — Статистичні результати експериментів

Параметр	Експеримент №1		Експеримент №2	
Вхідні умови проведення експерименту	$p = 17, n = 2, M = 8$		$p = 23, n = 2, M = 8$	
Алгоритм перетворення матриці	I	II	I	II
Об'єм сформованої вибірки N	83521	83521	279841	279841
N_{\max}	40319	40198	40319	40198
N_{\min}	0	0	0	0
Дисперсія, D	142498178,84	136109307,21	133524637,45	13542187,80
Відносна частота, W_{\max}	$3,5919 \cdot 10^{-5}$	$62,2597 \cdot 10^{-5}$	$2,50142 \cdot 10^{-5}$	$82,1895 \cdot 10^{-5}$
Відносна частота, W_{\min}	$2,3946 \cdot 10^{-5}$	$35,9191 \cdot 10^{-5}$	$2,1440 \cdot 10^{-5}$	$28,5877 \cdot 10^{-5}$
Кількість повторень ger_{\max}	3	52	7	230
Кількість повторень ger_{\min}	2	0	6	0
Медіана вибірки	19439	20012	19988	20162
Середнє значення вибірки	19514	19995	19996,96	20072,22

У результаті перетворення матриці A за першим алгоритмом отримано максимальну кількість повторів сформованих номерів перестановок у лексикографічному порядку їх слідування, $rep_{\max}^{A1} = 3$ у першому експерименті та $rep_{\max}^{A2} = 7$ у другому експерименті. Мінімальна кількість повторів сформованих номерів перестановок становить: $rep_{\min}^{A1} = 2$ у першому експерименті та $rep_{\min}^{A2} = 6$ у другому експерименті.

У результаті перетворення з використанням допоміжних матриць (другий алгоритм) отримано максимальну кількість повторів номерів перестановок $rep_{\max}^{B1} = 52$ у першому експерименті та $rep_{\max}^{B2} = 230$ у другому експерименті. Мінімальна кількість повторів номерів перестановок становить $rep_{\min}^{B1} = 0$ у першому експерименті та $rep_{\min}^{B2} = 0$ у другому експерименті.

Значення дисперсії отриманих номерів перестановок першого експерименту і обробкою матриці за першим алгоритмом становить $D_{A1} = 142498178,841$. Значення дисперсії першого експерименту і алгоритму обробки через допоміжні матриці становить $D_{B1} = 136109307,2089$.

3.6.4. Оцінка за критерієм Пірсона χ^2 результатів формування ключів-перестановок через квадратну матрицю

Отримане співвідношення $D_{A1} > D_{B1}$ вказує на те, що перший алгоритм має більші відхилення значень випадкової величини від центру розподілу, що є передумовою для більш рівномірного її розподілу на множині можливих значень. На гістограмі абсолютного розподілу перетворення матриці за другим алгоритмом (Рисунок 3.7) та (Рисунок 3.11) спостерігаються нулі – це означає, що відповідний номер перестановки не сформоване у вибірці.

Для перевірки нульової гіпотези H_0 про відповідність отриманих вибірок номерів перестановок у лексикографічному порядку їх слідування рівномірному розподілу на відрізку $[0; M! - 1]$ застосуємо критерій χ^2 Пірсона. Відповідно до [175] для коректного застосування критерія χ^2 отримана

вибірка має бути згрупована в класи. У праці [176] встановлено, що оптимальне число $r = r(N, \alpha)$ класів групування (інтервалів групування даних) за об'ємом вибірки N і рівнем значущості α для критерія χ^2 задається формулою

$$r(N, \alpha) = 2 \cdot 2^{1.2} \cdot \left(\frac{N-1}{\text{Norm}_{0;1;\alpha}} \right)^{0.4}, \quad (3.15)$$

де $\text{Norm}_{0;1;\alpha}$ – верхня межа стандартного нормального розподілу $\text{Norm}_{0;1}$.

Пізніше було встановлено [177], що без суттєвих втрат потужності критерію можна значення $r(N, \alpha)$ зменшити вдвоє і за великих N обрати

$$r(N, \alpha) = 2 \cdot 2^{0.2} \cdot \left(\frac{N-1}{\text{Norm}_{0;1;\alpha}} \right)^{0.4}. \quad (3.16)$$

Для $\alpha = 0.05$ значення $\text{Norm}_{0;1;0.05} = 1.645$.

Нижче наведено аналіз результатів експерименту 1 для першого алгоритму перетворення матриці.

Для $N = 83521$ і $\alpha = 0.05$ маємо $r(83521, 0.05) \approx 175.18$, отже кількість інтервалів групування $r = 175$. Довжина інтервалу групування дорівнює $H = \frac{40319}{175} \approx 230.3943$. Оберемо $H = 230.395$.

Тоді через $x_i^* = H \cdot i$, $0 \leq i \leq r$.

Через ν_i позначено абсолютні частоти інтервалів групування:

$$\nu_i = \sum_{x_{i-1}^* < x_j \leq x_i^*} N_j, \quad 0 \leq i \leq r, \quad \text{а} \quad \nu_1 = \sum_{x_0^* \leq x_j \leq x_1^*} N_j, \quad N_j - \text{абсолютна частота значення } x_j$$

на i -му інтервалі групування.

Якщо степінь вільності m розподілу χ_m^2 набуває великих значень, то в силу центральної граничної теореми цей розподіл наближається до нормального з параметрами $a = m$ і $\sigma^2 = 2m$:

$$\chi_m^2(x) \approx \text{Norm}_{m;2m}(x) \quad \text{для } m > 30. \quad (3.17)$$

Оскільки $Norm_{m;2m}(x) = Norm_{0;1}\left(\frac{x-m}{\sqrt{2m}}\right)$, то рівняння для верхньої α -межі розподілу $Norm_{m;2m}(x)$:

$$1 - \alpha = Norm_{m;2m}(x_\alpha) = Norm_{0;1}\left(\frac{x_\alpha - m}{\sqrt{2m}}\right) = Norm_{0;1}(t_\alpha). \text{ Тоді } \frac{x_\alpha - m}{\sqrt{2m}} = t_\alpha \text{ або}$$

$$x_\alpha = m + t_\alpha \sqrt{2m}. \quad (3.18)$$

Статистика критерія Пірсона для рівномірного розподілу має ступінь вільності $m = r - 1$. Якщо $\chi^2_{\alpha;m}$ – верхня α -межі розподілу χ^2 з m степенями вільності, то гіпотезу H_0 про неперервний рівномірний розподіл вибірки приймають на рівні значущості α , якщо

$$\chi^2_{\text{спост}} = \sum_{i=1}^r \frac{(v_i - Np_i)^2}{Np_i} < \chi^2_{\alpha;m}, \quad (3.19)$$

де p_i – теоретична відносна частота потрапляння досліджуваної випадкової величини в i -ий інтервал групування.

Варто зауважити, що:

$$1) \quad p_1 = P(0 \leq X \leq x_1^*) = P(x_{i-1}^* < X \leq x_{i1}^*) = p_i = \frac{1}{r}, \quad 2 \leq i \leq r. \text{ Для } N = 83521 \text{ і}$$

$$\alpha = 0.05 \text{ значення } p_i = \frac{1}{175}, \quad 1 \leq i \leq 175;$$

$$2) \text{ для всіх значень } i, \quad 1 \leq i \leq 175, \text{ виконано } Np_i = 83521 \cdot \frac{1}{175} \gg 10;$$

3) в силу співвідношень (3.17), (3.18) і (3.19) має місце

$$\chi^2_{\text{observ}} = \sum_{i=1}^r \frac{(v_i - N/r)^2}{N/r} = \frac{r}{N} \sum_{i=1}^r (v_i - N/r)^2 < \chi^2_{\alpha;r-1} \approx r - 1 + t_\alpha \sqrt{2} \sqrt{r-1}.$$

Таким чином, отримуємо правило перевірки:

$$\sum_{i=1}^r \left(v_i - \frac{N}{r} \right)^2 < \frac{N}{r} (r - 1 + t_\alpha \sqrt{2} \sqrt{r-1}). \quad (3.20)$$

Для $N = 83521$, $\alpha = 0.05$, $t_\alpha = 1.645$, $r = 175$ гіпотеза H_0 про рівномірність розподілу номерів перестановок на відрізку $[0; 40319]$ приймається за умови

$$\sum_{i=1}^{175} (v_i - 477.26)^2 < 97689.54. \quad (3.21)$$

Підставляючи в (3.21) отримані в результаті виконання експерименту 1 для першого алгоритму перетворення матриці абсолютні частоти v_i , отримаємо $\sum_{i=1}^{175} (v_i - 477.26)^2 = 603773,91$, що значно перевищує допустиме значення 97689.54.

Таблиця 3.2 демонструє результати застосування критерія Пірсона в вигляді формули (3.20) для результатів експериментів з (Таблиця 3.1).

Таблиця 3.2 — Результати перевірки гіпотези про рівномірність перестановок

Параметр	Експеримент №1		Експеримент №2	
Алгоритм перетворення матриці	I	II	I	II
Кількість інтервалів r	175		284	
Ширина інтервалу H	230,395		141,969	
Значення $\sum_{i=1}^r \left(v_i - \frac{N}{r} \right)^2$	603773,91	1973127,91	317923,08	350321441
Критичне значення $\frac{N}{r} \left(r - 1 + t_\alpha \sqrt{2} \sqrt{r - 1} \right)$	97689,54		317418,32	

Як можна бачити з таблиці, жоден з отриманих розподілів перестановок не відповідає рівномірному за критерієм χ^2 Пірсона. Разом з тим, для першого алгоритму перетворення матриці зі збільшенням відношення $\frac{p^{n^2}}{M!}$ така невідповідність зникає. Натомість, другий алгоритм перетворення матриці потребує додаткових досліджень для поліпшення статистичних властивостей.

3.7. Висновки

Визначено шість сімейств матриць із загальної лінійної групи $GL(2, \mathbb{Z}_p)$ порядку 2 над простим полем цілих чисел за модулем p з комутативною операцією множення. Визначено потужності множин для заданих сімейств.

Показано, що сімейство матриць $CGL_{b,k}(2, \mathbb{Z}_p)$, утворене множиною матриць $\left\{ t \cdot \begin{pmatrix} a & 1 \\ b & a+k \end{pmatrix}, t, a, b, k \in \mathbb{Z}_p, t \neq 0, a(a+k) - b \neq 0 \right\}$ та доповнене множиною матриць $\left\{ s \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, s \in \mathbb{Z}_p, s \neq 0 \right\}$, утворює абелеву групу відносно операції множення. Порядок отриманої групи становить: $p^2 - 1$.

Визначено комутативне сімейство матриць $CGL_{b,k}(2, \mathbb{Z}_p)$, що одночасно діагоналізується над полем $F_{p^2} = \mathbb{Z}_p[\sqrt{D}]$.

Показано, що сімейство матриць $F_{b,k} = CGL_{b,k}(2, \mathbb{Z}_p) \cup \Theta$, де $\Theta = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$, утворює поле Галуа порядку p^2 із звичайними операціями множення та додавання матриць. Відповідно, $CGL_{b,k}(2, \mathbb{Z}_p)$ є мультиплікативною циклічною групою. Використаний підхід для знаходження скінченного поля матриць порядку 2, може бути розширений для вивчення квадратних матриць вищих порядків, наприклад для матриць третього порядку.

Результати теоретичного обґрунтування створили передумову для інтегрування методів узгодження ключових елементів у вигляді матриць з факторіальним кодуванням даних, що використовує для переносу інформації перестановки.

Досліджено використання матриць для узгодження ключів-перестановок. Для цього створено та реалізовано програмну модель перетворення квадратної матриці A в перестановку π .

Запропоновано та статистично перевірено два алгоритми перетворення матриці:

- шляхом представлення матриці A в десятковій та факторіальній системах числення;
- шляхом використання допоміжних квадратних матриць.

Перший алгоритм перетворення передбачає поетапне перетворення елементів матриці в коефіцієнти позиційної (десяткової та факторіальної) систем числення. Другий алгоритм перетворення передбачає почергове множення зліва матриці A на всі допоміжні матриці з наступним формуванням факторіальних коефіцієнтів з елементів матриць-добутків.

Проведені два експерименти за різних значень розмірності скінченного поля Z_p елементів матриці: $p=17$ та $p=23$. Побудовані гістограми розподілів абсолютних та відносних частот отриманих перестановок на виході програмної моделі.

Алгоритм перетворення шляхом представлення матриці A в десятковій та факторіальній системах числення за великих $\frac{p^{n^2}}{M!}$ показав можливість застосування для формування рівномірного розподілу можливих перестановок і може використовуватись як засіб формування ключів у системах з факторіальним кодуванням.

Другий алгоритм перетворення з використанням допоміжних матриць не забезпечує рівномірний розподіл сформованих перестановок та потребує подальших досліджень.

Основні результати дослідження побудови скінчених полів матриць порядку 2 та дослідження алгоритмів перетворення квадратної матриці в перестановку шляхом побудови програмної моделі представлено в [178], [179], [180]

4. ПРАКТИЧНІ АСПЕКТИ РЕАЛІЗАЦІЇ СИСТЕМ ІНФОРМАЦІЙНОГО ОБМІНУ З НЕРОЗДІЛЬНИМ ФАКТОРІАЛЬНИМ КОДУВАННЯМ ДАНИХ

4.1. Вступ

Створення імітаційної моделі у комп'ютерному середовищі надає змогу суттєво скоротити витрати часу та ресурсів, необхідних для розробки й тестування протоколів інформаційного обміну. Зокрема, це стосується протоколів, що базуються на нероздільному факторіальному кодуванні. Використання такої моделі дозволяє провести верифікацію теоретичних оцінок достовірності передавання даних, а також забезпечити ефективне налаштування інформаційних систем обміну, що використовують факторіальні коди та перевірки методів на їх основі.

З метою практичної реалізації розроблених систем інформаційного обміну створено макетні зразки приймально-передавальних пристроїв на основі контролера nRF52840, що працює в ISM-діапазоні. Компоненти системи реалізовані у вигляді компактних модулів прототипу з можливістю інтеграції в IoT-рішення.

Окрім того, розглянуто можливості прискорення операцій над перестановками за допомогою графічних процесорів. Використання обчислювальних можливостей CUDA дозволяє значно підвищити швидкість обчислення операцій множення перестановок, що є критично важливим для реалізації криптографічних протоколів на їх основі. Досліджено ефективність застосування GPU порівняно з CPU, виявлено залежність продуктивності від кількості паралельних обчислень та оцінено перспективи використання таких рішень у задачах захищеного інформаційного обміну.

У четвертому розділі показано такі практичні результати:

- розроблено імітаційну модель системи інформаційного обміну з нероздільним факторіальним кодуванням даних;

- створено макетні зразки приймально-передавальних пристроїв захищеного інформаційного обміну текстовими повідомленнями через радіоканал ISM діапазону з використанням перестановок;
- підтверджено ефективність застосування графічних прискорювачів для криптографічних операцій над перестановками.

4.2. Імітаційна модель системи інформаційного обміну

Для побудови імітаційної моделі системи інформаційного обміну на основі НФКД використано програмне забезпечення Matlab і Simulink.

Для створення імітаційної моделі системи інформаційного обміну перестановками в каналі зв'язку з шумами виконано наступні завдання:

- визначено структуру моделі та реалізовано її в комп'ютерному середовищі;
- визначено перелік експериментів для перевірки роботи моделі;
- реалізовано експериментальні дослідження передавання перестановок каналами різної якості;
- виконано порівняння отриманих результатів з теоретичними.

4.2.1. Структура імітаційної моделі

Структурна схема імітаційної моделі системи передавання інформації з нероздільним факторіальним кодуванням даних має вигляд, наведений на (Рисунок 4.1).

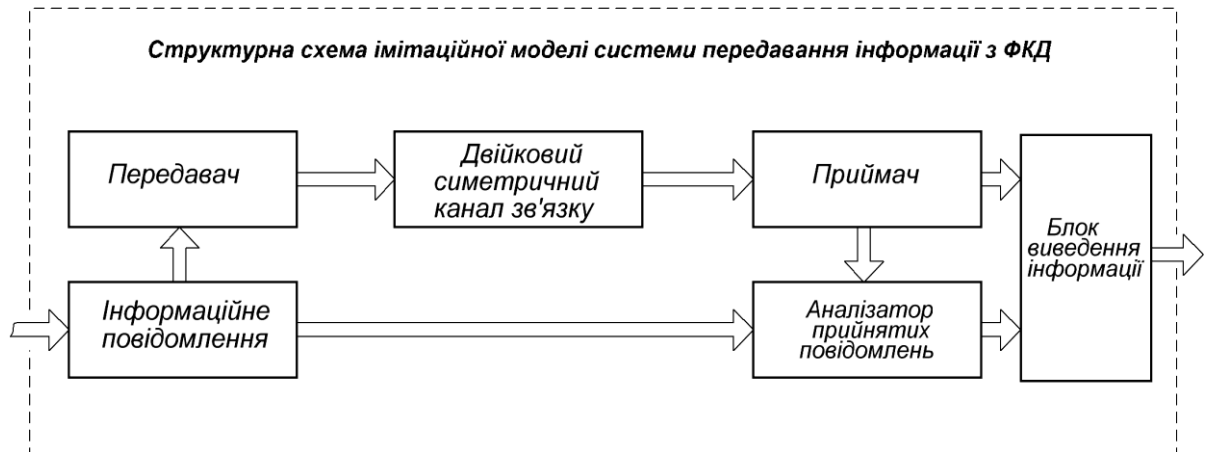


Рисунок 4.1 – Структурна схема імітаційної моделі

Функціональні складові побудованої імітаційної моделі в програмному середовищі наведено нижче (Рисунок 4.2).

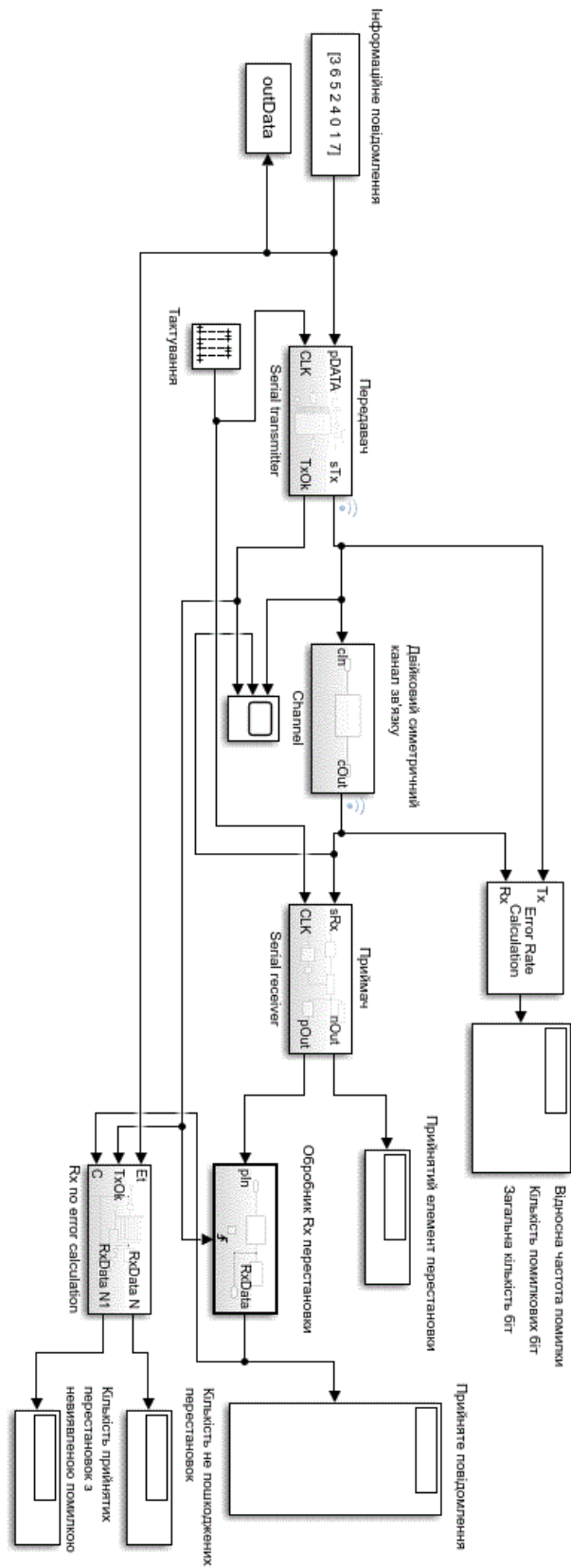


Рисунок 4.2 – Загальна структура реалізованої імітаційної моделі в програмному середовищі

Інформаційне повідомлення – повідомлення, бієктивно відображене на перестановку. За довжини перестановки M потужність множини всіх перестановок дорівнює $M!$, а тому передавач може сформувати до $M!$ різних повідомлень.

Двійковий симетричний канал зв'язку (Channel) – канал передавання двійкових даних з однаковою перехідною ймовірністю p_0 .

Передавач (serial transmitter) – виконує послідовне, побітове передавання перестановки в двійковий симетричний канал зв'язку. Вхідні сигнали: $pData$ – перестановка-повідомлення; CLK – сигнал тактування. Вихідні сигнали: sTx – біти інформаційного повідомлення, з'єднаний з каналом передавання; $TxOk$ – сигнал завершення передавання всіх елементів перестановки в канал зв'язку. Сигнал формується після передавання останнього біту інформаційного повідомлення.

Приймач (serial reciever) – виконує послідовне, побітове приймання перестановки з каналу зв'язку. Вхідні сигнали: sRx – перестановка-повідомлення, яка послідовно надходить з каналу зв'язку; CLK – сигнал тактування. Вихідні сигнали: $nOut$ – елемент перестановки, отриманий приймачем; $pOut$ – перестановка, отримана приймачем.

Обробник Rx перестановки – приводить отриману з двійкового симетричного каналу зв'язку перестановку до початкового вигляду. Наприклад у результаті передавання перестановки

$$\pi = \{\pi_0, \pi_1, \dots, \pi_{M-1}\}, \quad (4.1)$$

передавачем у двійковий симетричний канал *приймач* отримує символи перестановки у зворотній послідовності – $\pi = \{\pi_M, \pi_{M-1}, \dots, \pi_0\}$.

Лічильник прийнятих перестановок (Rx no error calculation) – рахує кількість перестановок, що співпадають з відправленими, а також фіксує кількість перестановок з невиявленою помилкою. Вхідні сигнали: Et – еталонне значення переданої перестановки (*перестановка що передається*); S – отримана приймачем перестановка; $TxOk$ – сигнал завершення передавання

всіх елементів перестановки в канал зв'язку. Вихідні сигнали: Rx Data N – кількість непошкоджених перестановок, Rx Data N 1 – кількість перестановок з невиявленою помилкою.

Лічильник неправильно прийнятих бітів (Error Rate Calculation) – рахує кількість помилкових бітів на виході двійкового симетричного каналу зв'язку. Вхідні сигнали: Tx сигнал з виходу приймача, ідентичний sTx; Rx – сигнал на виході каналу (вході приймача), ідентичний cOut, sRx.

Блок тактування – формує сигнал CLK для функціонування моделі.

Дисплеї відображення даних та *осцилограф* слугують для виводу проміжних і остаточних сигналів при моделюванні.

4.2.2. Налаштування імітаційної моделі

Налаштування моделі задаються в меню Modeling → Model setting, (Рисунок 4.3).

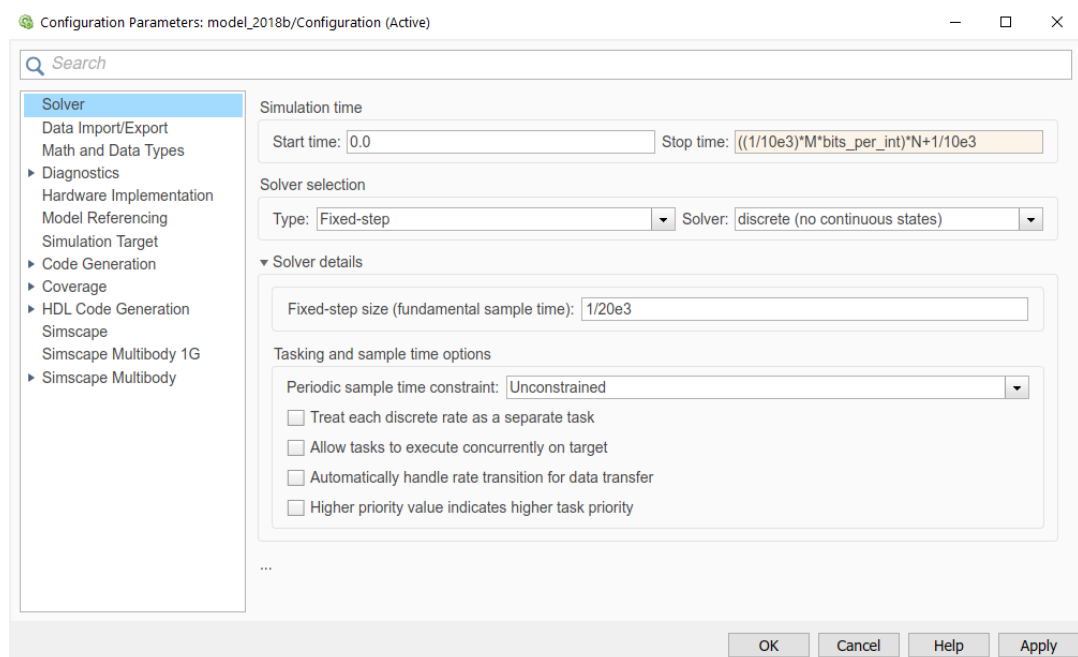


Рисунок 4.3 – Параметри налаштування симуляції моделі

До основних параметрів налаштування моделі відносяться (рисунок 4.3):

simulation (stop time) – кінцевий час для зупинки симуляції (моделювання), значення обраховується за формулою:

$$Stop\ time = \left(\frac{1}{f_{txrx}} \cdot M \cdot l_r \right) \cdot N + \frac{1}{f_{txrx}}, \quad (4.2)$$

де:

f_{txrx} – частота передавання (у роботі прийнято $f_{txrx} = 10^3 \text{ Гц}$);

M – довжина перестановки (у роботі прийнято $M = 8$);

l_r – кількість бітів для кодування одного елементу перестановки,

$[\log_2 M] = 3$;

N – кількість тестових передавань перестановки;

$\frac{1}{f_{txrx}}$ – час для оцінки отримання даних.

solver selection (type: fixed-step) – фіксований крок симуляції;

solver selection (solver: discrete) – дискретний обробник моделі;

fundamental sample time – мінімальний проміжок часу в режимі симуляції. З метою отримання певної частоти передавання символів f_{txrx} параметр мінімального проміжку часу має бути в два рази вищий – $2 \cdot f_{txrx}$.

Для синхронізації передавача та приймача введено блок тактування. Приклад налаштування блоку тактування зображено нижче.

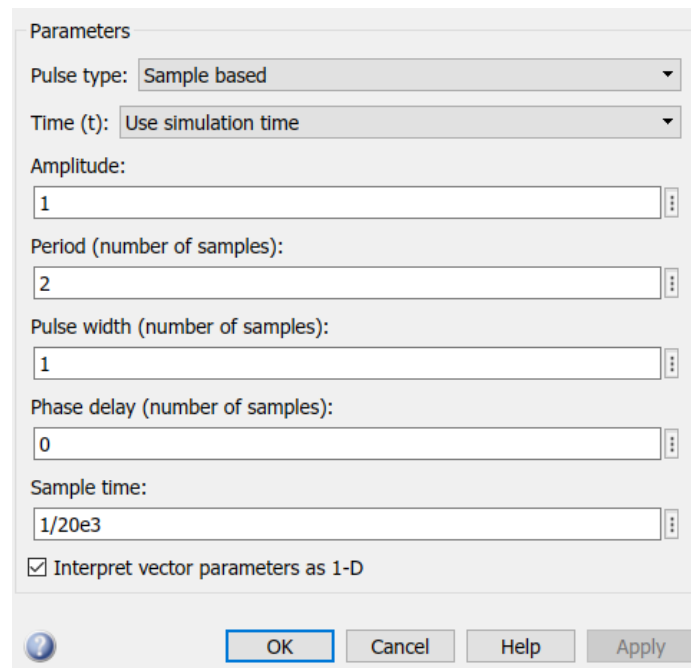


Рисунок 4.4 – Налаштування блоку тактування

4.2.3. Підсистеми імітаційної моделі

Передавач (serial transmitter) виконує послідовне, побітове передавання даних. Передавач містить (Рисунок 4.5): блок перетворення даних (integer to bit converter), лічильник тактування (counter), мультиплексор, блок перетворення даних (boolean), блок формування затримки сигналу (delay), блок відображення (display), блок порівняння (compare to constant), вхідні та вихідні порти.

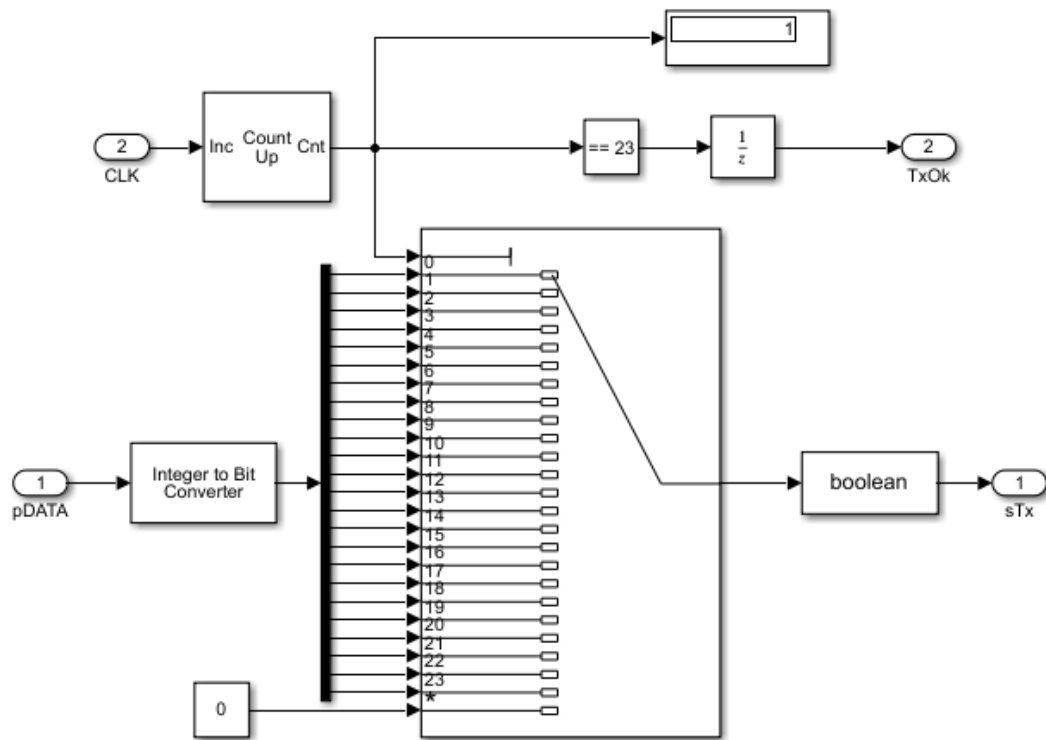


Рисунок 4.5 – Передавач (serial transmitter)

Підсистема отримує сигнал тактування ззовні через порт CLK на лічильник, налаштування лічильника представлено на (Рисунок 4.6). На вхідний порт pDATA надходить перестановка що передається. За визначених M і l_r кількість біт n , відповідно, тактових імпульсів для передавання однієї перестановки становить:

$$n = M \cdot l_r = 8 \cdot 3 = 24. \quad (4.3)$$

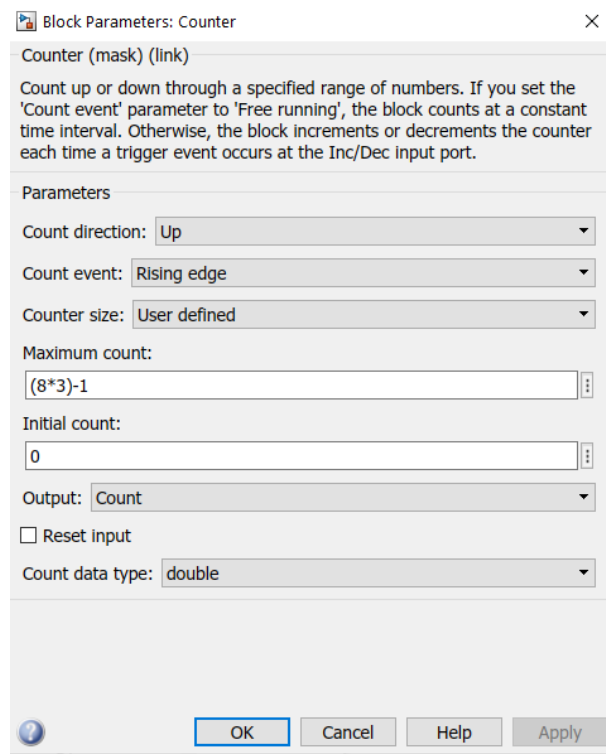


Рисунок 4.6 – Налаштування блоку лічильника

Перестановка надходить на блок перетворювача *integer to bit*, де кожен елемент перестановки перетворюється в двійковий вигляд із заданою довжиною кодової комбінації l_r (Рисунок 4.7).

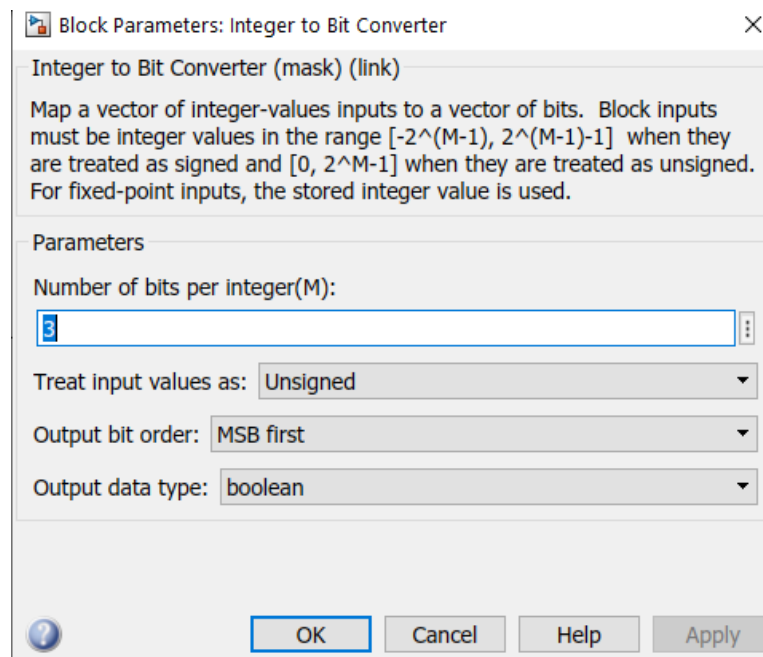


Рисунок 4.7 – Налаштування блоку *integer to bit*

Послідовність біт перестановки надходить через шину до мультиплексора на 24 входи. Вихід лічильника з'єднано з адресним входом мультиплексора. У залежності від кількості обрахованих імпульсів лічильником, кожен із входів мультиплексора з'єднується з його виходом. Сигнал даних з мультиплексора побітово передається на вихідний порт sTx під дією сигналу тактування. Таким чином реалізовано послідовна передача бітів в канал. Вихідний сигнал TxOk формується після передавання останнього, 24-го (для $l_r = 3$, див. формула (4.2)) біту на вихідний порт sTx.

Приймач (*serial reciever*) отримує бітову послідовність, передану каналом зв'язку, декодує її та формує вихідне інформаційне повідомлення. Приймач містить (Рисунок 4.8): n -бітовий запам'ятовувальний пристрій (*serial-in parallel-out bits*), детектор елементу перестановки, підсистема перетворення двійкового представлення в десяткове (*convert to int*), блок формування перестановки (Рисунок 4.9), порти вхідних і вихідних даних.

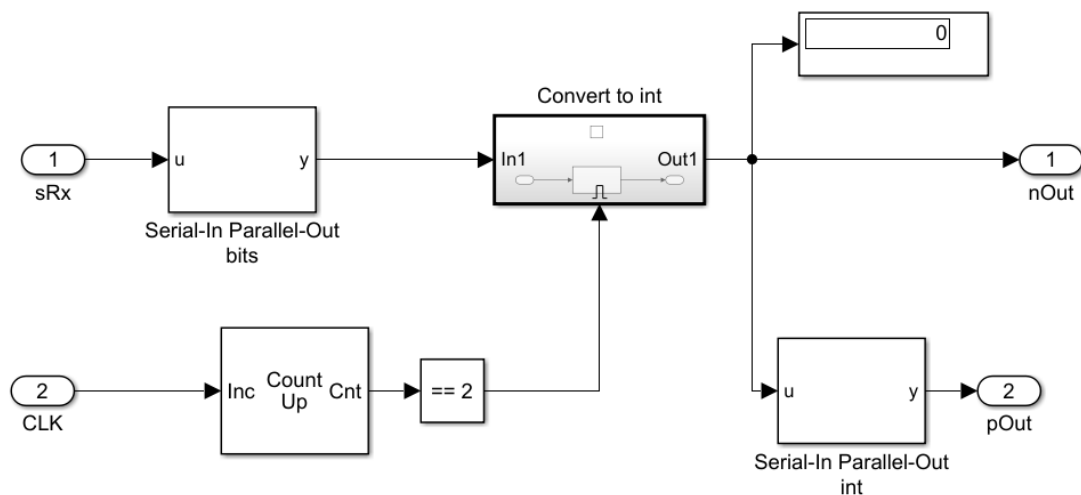


Рисунок 4.8 – Приймач (serial reciever)

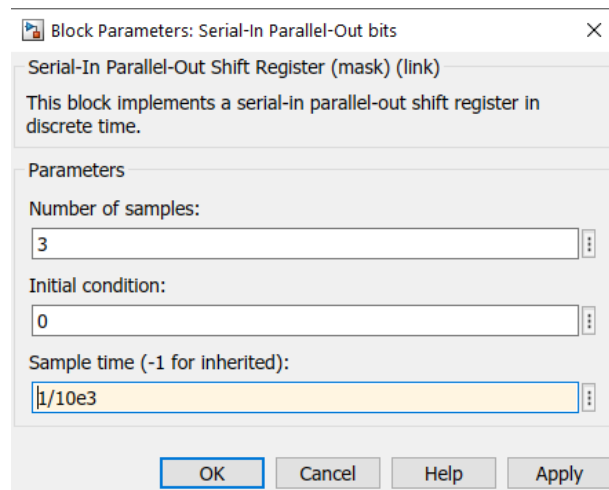


Рисунок 4.9 – Налаштування блоку формування перестановки

Сигнал тактування CLK та дані sRx потрапляють на n -бітовий запам'ятовувальний пристрій. Сигнал тактування CLK також надходить до детектору елементу перестановки – лічильника. Лічильник обраховує кількість бітів, що надійшла через вхідний порт sRx, і формує сигнал дозволу перетворення даних з двійкового в десяткове представлення для підсистеми *convert to int* (Рисунок 4.10). Налаштування блоку *bit to integer converter* представлено на (Рисунок 4.11).

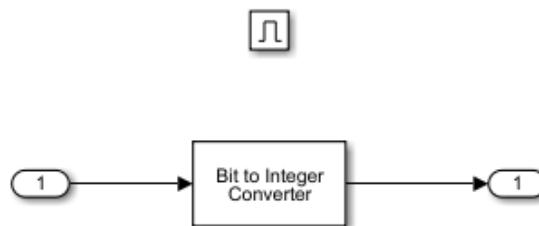


Рисунок 4.10 – Підсистема перетворення даних з двійкового в десяткове представлення

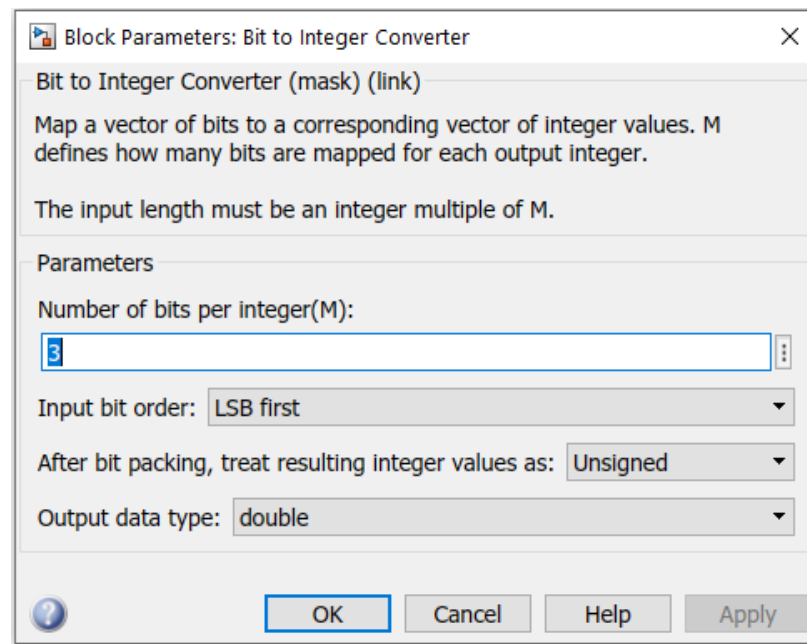


Рисунок 4.11 – Налаштування підсистеми перетворення даних з двійкового в десяткове представлення

Обробник *Rx* перестановки впорядковує прийняті елементи перестановки. Отримана підсистемою перестановка приводиться до початкового вигляду за (4.1), алгоритм упорядкування реалізовано окремою функцією Matlab з вхідними даними u та вихідними y . Підсистема також виконує запис до зовнішнього файлу прийняті перестановки. Підсистема є тригерною, тобто спрацьовує за сигналом $TxOk$.

Обробник *Rx* перестановки містить (Рисунок 4.12): функцію впорядкування елементів перестановки (*pResort*), блок збереження даних до зовнішнього файлу *inData*, вхідні та вихідні порти.

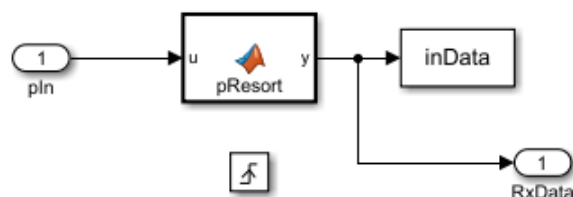


Рисунок 4.12 – Обробник *Rx* перестановки

Лічильник прийнятих перестановок (Rx no error calculation) порівнює відправлену перестановку з отриманою, веде підрахунок кількості перестановок які ідентичні вхідному інформаційному повідомленню, окремо фіксує отримані перестановки з невиявленою помилкою, які є перестановками, проте не ідентичні відправленому інформаційному повідомленню.

Лічильник містить (Рисунок 4.13): блок порівняння, аналізатор елементів перестановок, лічильник перестановок не пошкоджених помилкою, підсистему фіксації невиявлених пошкоджених перестановок (Рисунок 4.14), вхідні та вихідні порти.

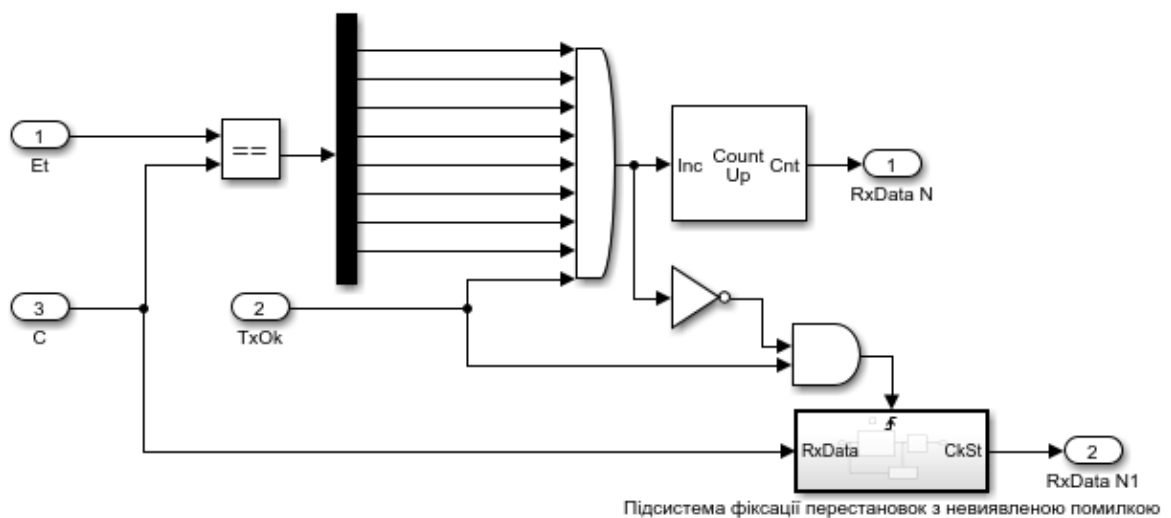


Рисунок 4.13 – Лічильник прийнятих перестановок (Rx no error calculation)

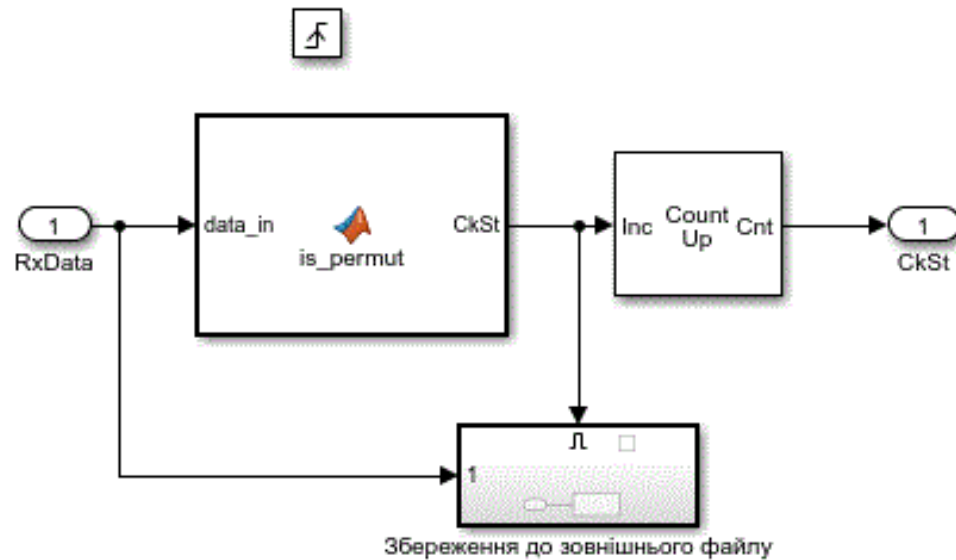


Рисунок 4.14 – Підсистема фіксації перестановок з невиявленою помилкою

На вхід підсистеми *Rx no error calcuation* при надходженні двох перестановок: вихідна – та що передається передавачем (порт Et) та перестановка на виході двійкового симетричного каналу (порт C). Ці перестановки потрапляють до блоку порівняння, в якому кожен елемент перестановки оброблюється за заданою умовою:

$$\begin{aligned}
 OK &= [C_M, C_{M-1}, \dots, C_1] \& \\
 &\& [Et_M, Et_{M-1}, \dots, Et_1] \& [TxOk] = \\
 &= \left[(C_M) \& (Et_M), (C_{M-1}) \& (Et_{M-1}), \dots \right] \& \\
 &\quad \left[\dots, (C_1) \& (Et_1) \right] \& \\
 &\quad \& (TxOk).
 \end{aligned}$$

(4.4)

де:

OK – логічний сигнал що потрапляє на лічильник перестановок, не пошкоджених помилкою. Сигнал формується за умови співпадіння всіх

елементів перестановки, отриманих приймачем, з початковими елементами вхідної перестановки;

C_M – перестановка на виході двійкового симетричного каналу зв'язку;

Et – вихідна перестановка для передавання.

У результаті аналізу заданої блоком порівняння умови (елементи повинні співпадати) формуються логічні сигнали: логічна «1», якщо елементи задовольняють умову, логічний «0», якщо елементи не задовольняють умову. Приклад функціонування блоку порівняння за

$$\begin{aligned}
 OK &= [C_M, C_{M-1}, \dots, C_1] \& \\
 &\& [Et_M, Et_{M-1}, \dots, Et_1] \& [TxOk] = \\
 &= \left[(C_M) \& (Et_M), (C_{M-1}) \& (Et_{M-1}), \dots \right] \& \\
 &\& (TxOk).
 \end{aligned}$$

(4.4)

наведено на схемі зображеній на (Рисунок 4.15).

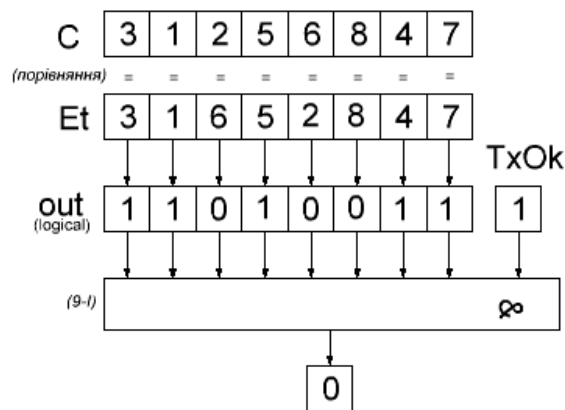


Рисунок 4.15 – Схема формування сигналу підрахунку кількості перестановок, пошкоджених каналною помилкою

4.2.4. Опис алгоритму підрахунку перестановок, не пошкоджених помилкою

До аналізатора елементів перестановок (логічний елемент AND на 9 входів) надходить отриманий вектор логічних значень (Рисунок 4.15, *out*). У результаті виконання логічної умови (4.4)

$$\begin{aligned}
 OK &= [C_M, C_{M-1}, \dots, C_1] \& \\
 &\& [Et_M, Et_{M-1}, \dots, Et_1] \& [TxOk] = \\
 &= \left[(C_M) \& (Et_M), (C_{M-1}) \& (Et_{M-1}), \dots \right] \& \\
 &\left[\dots, (C_1) \& (Et_1) \right] \& \\
 &\& (TxOk).
 \end{aligned}$$

(4.4) та наявності сигналу передавання останнього біту поточної перестановки через порт TxOk, формується сигнал лічильнику про те, що перестановку з M елементів передано вірно. За умови неспівпадіння хоча б одного елементу перестановки сигнал не формується (Рисунок 4.15 демонструє ситуацію виявлення перестановки пошкоджених помилкою). Кількість отриманих сигналів після аналізатора елементів перестановок фіксуються лічильником, а результат підрахунку виводиться на дисплей через вихідний порт RxData N.

4.2.5. Підсистема фіксації перестановок з невиявленою помилкою.

Склад підсистеми (Рисунок 4.14): функція *is_permut*, лічильник перестановок з невиявленою помилкою, підсистема збереження зафіксованих перестановок до зовнішнього файлу, вхідні вихідні порти. Функція *is_permut* виконує роль аналізатора даних, отриманих з входу RxData. Функція повертає логічну одиницю, якщо вхідна послідовність є перестановкою. Кількість сформованих сигналів фіксується лічильником, значення лічильника надходить на вихідний порт RxData N1.

Підсистема фіксації перестановок з невиявленою помилкою отримує сигнал на виконання від підсистеми *Rx no error calculation* за таких умов:

- отримана перестановка через вхідний порт С не співпадає з відправленою перестановкою (вхідний порт Et);
- відбулося передавання останнього біту в канал зв'язку (сигнал TxOk).

Двійковий симетричний канал зв'язку (*Channel*) містить (Рисунок 4.16): блок binary symmetric channel (BSC), вхідні та вихідні порти.

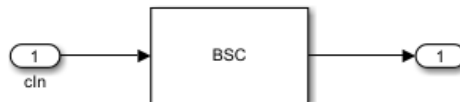


Рисунок 4.16 – Двійковий симетричний канал зв'язку (*Channel*)

До головних налаштувань блоку BSC входять (Рисунок 4.17): імовірність помилки (*Error probability*) від 0 до 1; початкове «зерно» (*initial seed*) використовується для генерації псевдовипадковості.

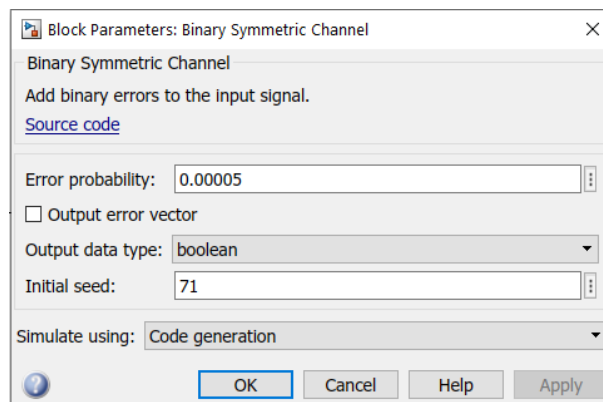


Рисунок 4.17 – Налаштування binary symmetric channel

4.2.6. Результати роботи імітаційної моделі інформаційного обміну перестановками

З метою оцінювання правильності роботи всіх компонентів моделі системи передавання перестановок двійковим симетричним каналом з визначеною величиною бітової помилки в каналі передавання виконано

імітаційне моделювання за визначеними параметрами: довжина перестановки $M = 8$, кількість ітерацій передавання перестановки $N = 10000$, імовірності бітових помилок для кожного експерименту: $p_0 = 0.05$ (експеримент №1), $p_0 = 0.1$ (експеримент №2), $p_0 = 0.15$ (експеримент №3). У результаті зібрано результати моделювання, що показані нижче, та проведено порівняння отриманих результатів з теоретичними оцінками за визначених умов.

Опис та результати експерименту №1. Імовірність бітової помилки $p_0 = 0.05$, *initial seed* 666, вхідна перестановка $\pi = [3, 6, 5, 2, 4, 0, 1, 7]$.

Кількість перестановок, отриманих без пошкодження помилкою, дорівнює $N_{noerr}(0.05) = 2929$. Кількість перестановок з невиявленою помилкою склала $N_{ud}(0.05) = 110$. Кількість перестановок, пошкоджених помилкою та перетворених у не перестановку, дорівнює $N_{det}(0.05) = 6961$ (Рисунок 4.18).

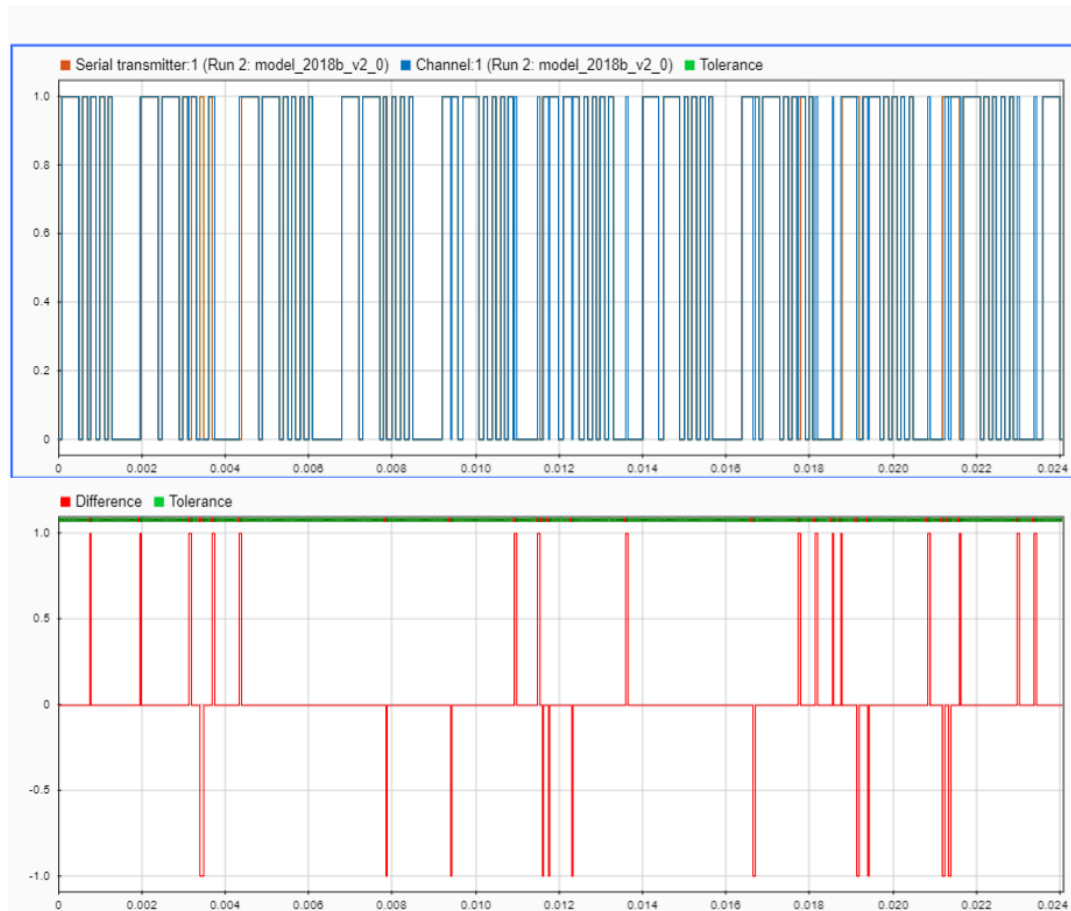


Рисунок 4.18 – Графік Data Inspection за ймовірності бітової помилки $p_0 = 0.05$ (наведено перші 10 блоків інформаційного повідомлення)



Рисунок 4.19 – Результати передавання бітів каналом зв'язку

Дані, зображені на (Рисунок 4.18), показують вміст каналу зв'язку та значення відхилення сигналу на виході відносно входу. Відповідно верхній графік демонструє сигнали на вході та на виході каналу зв'язку, при цьому вісь абсцис позначає час симуляції. Відмінності в сигналах на вході та на виході каналу зв'язку зафіксовано на нижньому графіку. Відхилення приймає значення з множини $\{-1;1\}$. Значення «-1» означає присутність логічного нуля

в отриманих даних замість логічної одиниці, «1» – присутність логічної одиниці в отриманих даних, де повинен бути логічний нуль.

Рисунок 4.19 демонструє відомості про статистичні показники передавання під час симуляції. У результаті проведення експерименту 1 маємо: відносна частота появи бітової помилки $w(0.05)=0.0491$, кількість помилкових бітів $1.179 \cdot 10^4$, загальна кількість переданих і прийнятих бітів $2.4 \cdot 10^5$.

Абсолютне відхилення значення відносної частоти появи бітової помилки від заданої в параметрах (Рисунок 4.17) p_0 з імовірністю $\gamma=0.95$ не повинно виходити за межі:

$$|w(p_0) - p_0| \leq \varepsilon = 1.96 \sqrt{\frac{p_0(1-p_0)}{N}}. \quad (4.5)$$

Іншими словами, відносна частота появи бітової помилки з імовірністю $\gamma=0.95$ повинна потрапляти у відрізок:

$$p_0 - 1.96 \sqrt{\frac{p_0(1-p_0)}{N}} \leq w(p_0) \leq p_0 + 1.96 \sqrt{\frac{p_0(1-p_0)}{N}} \quad (4.6)$$

Для $p_0=0.05$ вираз (4.6) отримує вигляд: $0.0457 \leq w(0.05) \leq 0.0543$, а значення відносної частоти $w(0.05)=0.0498$ потрапляє в цей відрізок. Відносна частота перестановок з виявленою помилкою в експерименті 1 склала $W_{det}(0.05) = N_{det}(0.05)/N = 0.6961$. Відносна частота перестановок з невиявленою помилкою – $W_{ud}(0.05) = N_{ud}(0.05)/N = 0.0110$.

Опис та результати експерименту №2. Імовірність бітової помилки $p_0=0.1$, *initial seed* 53, вхідна перестановка $\pi=[3,6,5,2,4,0,1,7]$. Кількість перестановок, отриманих без пошкодження помилкою, дорівнює $N_{noerr}(0.1)=768$. Кількість перестановок з невиявленою помилкою склала $N_{ud}(0.1)=139$. Кількість перестановок, пошкоджених помилкою та перетворених у не перестановку, дорівнює $N_{det}(0.1)=9093$, (Рисунок 4.20).

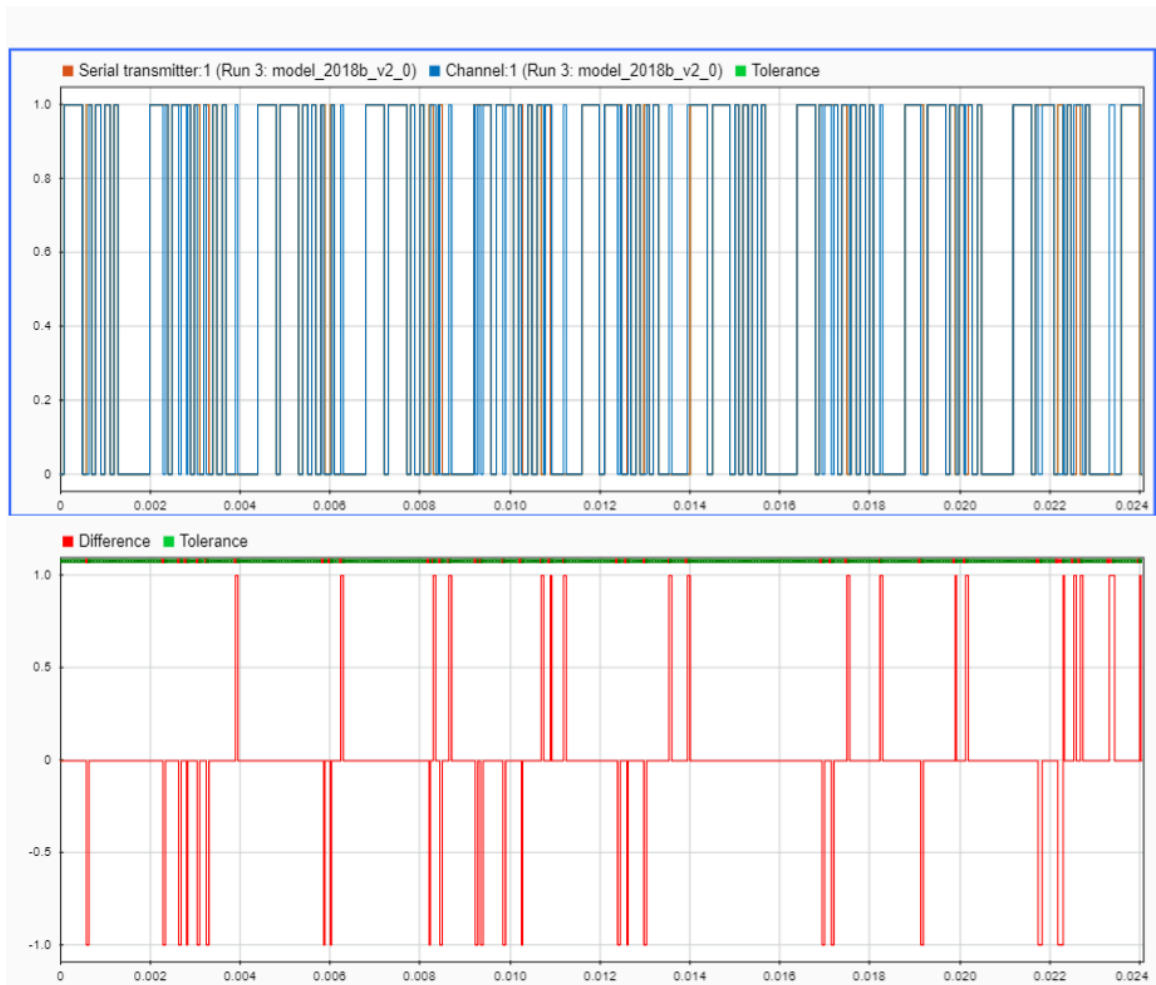


Рисунок 4.20 – Графік Data Inspection за ймовірності бітової помилки $p_0 = 0.1$ (наведено перші 10 блоків інформаційного повідомлення)

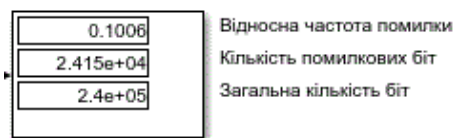


Рисунок 4.21 – Результати передавання бітів каналом зв'язку

Для $p_0 = 0.1$ вираз (4.6) отримує вигляд: $0.0941 \leq w(0.1) \leq 0.1059$, а значення відносної частоти $w(0.1) = 0.1006$ потрапляє в цей відрізок. Відносна частота перестановок з виявленою помилкою в експерименті 2 склала $W_{det}(0.1) = N_{det}(0.1)/N = 0.9093$. Відносна частота перестановок з невиявленою помилкою – $W_{ud}(0.1) = N_{ud}(0.1)/N = 0.0139$.

Опис та результати експерименту №3. Імовірність бітової помилки $p_0 = 0.15$, initial seed 74, вхідна перестановка $\pi = [3, 6, 5, 2, 4, 0, 1, 7]$. Кількість перестановок, отриманих без пошкодження помилкою, дорівнює $N_{noerr}(0.15) = 215$. Кількість перестановок з невиявленою помилкою склала $N_{ud}(0.15) = 94$. Кількість перестановок, пошкоджених помилкою та перетворених у не перестановку, дорівнює $N_{det}(0.15) = 9691$, (рисунок 4.22).

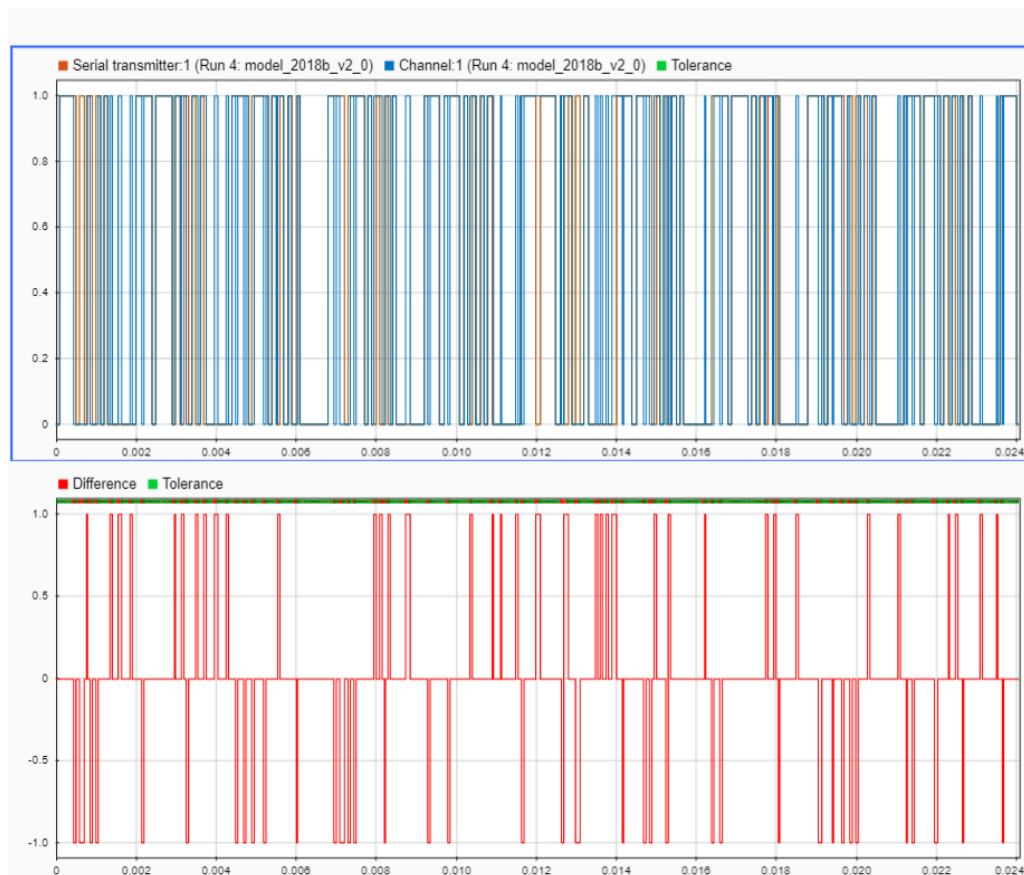


Рисунок 4.22 – Графік Data Inspection за ймовірності бітової помилки $p_0 = 0.15$ (наведено перші 10 блоків інформаційного повідомлення)

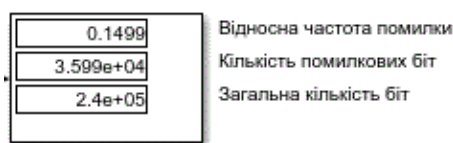


Рисунок 4.23 – Результати передавання бітів каналом зв'язку

Для $p_0 = 0.15$ вираз (4.6) отримує вигляд: $0.1430 \leq w(0.15) \leq 0.1570$, а значення відносної частоти $w(0.15) = 0.1499$ потрапляє в цей відрізок. Відносна частота перестановок з виявленою помилкою в експерименті 3 склала $W_{det}(0.15) = N_{det}(0.15)/N = 0.9691$. Відносна частота перестановок з невиявленою помилкою – $W_{ud}(0.15) = N_{ud}(0.15)/N = 0.0094$.

4.2.7. Оцінка роботи імітаційної моделі

Побудованою імітаційною моделлю проведено дослідження впливу каналної помилки на інформаційні повідомлення під час їхнього передавання. У результаті проведення 3-х експериментів отримано такі результати відносної частоти пошкодження перестановки та перетворення її в не перестановку (виявлення помилки в отриманій перестановці): експеримент 1 – $W_{det}(0.05) = 0.6961$ за $p_0 = 0.05$, експеримент 2 – $W_{det}(0.1) = 0.9093$ за $p_0 = 0.1$, експеримент 3 – $W_{det}(0.15) = 0.9691$ за $p_0 = 0.15$.

Відносна частота не виявленої в отриманій перестановці помилки, дорівнює: $W_{ud}(0.05) = 0.0110$, $W_{ud}(0.1) = 0.0139$, $W_{ud}(0.15) = 0.0094$.

Відносна частота отримання перестановки без помилок, дорівнює: $W_{noerr}(0.05) = 0.2929$, $W_{noerr}(0.1) = 0.0768$, $W_{noerr}(0.15) = 0.0215$.

Імовірність отримання перестановки без помилок дорівнює

$$Q(p_0) = (1 - p_0)^{24} \quad (4.7)$$

Для $p_0 \in \{0.05, 0.1, 0.15\}$ ця ймовірність набуває значень $Q(0.05) = 0.2920$, $Q(0.1) = 0.0798$, $Q(0.15) = 0.0202$.

Абсолютне відхилення значення відносної частоти отримання перестановки без помилок від теоретично визначеної ймовірності (4.7) з імовірністю $\gamma = 0.95$ не повинно виходити за межі:

$$|W_{noerr}(p_0) - Q(p_0)| \leq E = 1.96 \sqrt{\frac{Q(p_0)(1 - Q(p_0))}{N}} \quad (4.8)$$

Іншими словами, відносна частота отримання перестановки без помилок з імовірністю $\gamma = 0.95$ повинна потрапляти в відрізок

$$\begin{cases} W_{noerr}(p_0) \geq Q(p_0) - 1.96 \sqrt{\frac{Q(p_0)(1-Q(p_0))}{N}}, \\ W_{noerr}(p_0) \leq Q(p_0) + 1.96 \sqrt{\frac{Q(p_0)(1-Q(p_0))}{N}}. \end{cases} \quad (4.9)$$

Для $p_0 \in \{0.05, 0.1, 0.15\}$ вираз (4.9) отримує вигляд:

$$0.2831 \leq W_{noerr}(0.05) \leq 0.3009, \quad 0.0745 \leq W_{noerr}(0.1) \leq 0.0851,$$

$0.0175 \leq W_{noerr}(0.15) \leq 0.0230$. Як видно, отримані експериментальні показники лежать у визначених межах.

Довірчий інтервал для ймовірності невиявленої помилки в перестановці за відомою відносною частотою $W_{ud}(p_0)$ з надійністю $\gamma = 0.95$

$$\begin{cases} P_{ud}(p_0) \geq W_{ud}(p_0) - 1.96 \sqrt{\frac{W_{ud}(p_0)(1-W_{ud}(p_0))}{N}}, \\ P_{ud}(p_0) \leq W_{ud}(p_0) + 1.96 \sqrt{\frac{W_{ud}(p_0)(1-W_{ud}(p_0))}{N}}. \end{cases} \quad (4.10)$$

Таким чином, для $p_0 \in \{0.05, 0.1, 0.15\}$ і $\gamma = 0.95$ маємо:

$$0.0090 \leq P_{ud}(0.05) \leq 0.0130, \quad 0.0116 \leq P_{ud}(0.1) \leq 0.0162,$$

$$0.0075 \leq P_{ud}(0.15) \leq 0.0113.$$

У реальних системах наявність невиявлених помилок при отриманні повідомлень може призводити до аварійних ситуацій (стоп програми, некоректне розпізнавання даних) на стороні приймача. За високої ймовірності бітової помилки в каналі зв'язку отримані оцінки ймовірності невиявленої помилки факторіальним кодом з відновленням даних за перестановкою можуть не задовольняти поставленим вимогам і тому вимагати додаткових методів підвищення достовірності.

4.3. Побудова прототипу захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону

У цій частині розділу викладено особливості використання перестановок для створення системи інформаційного обміну текстовими даними при реалізації протоколу передавання текстових повідомлень із використанням нероздільного факторіального кодування даних. У результатах створена система інформаційного обміну текстовими даними та реалізована у вигляді макетного зразку.

Для створення цієї системи, відповідного програмного забезпечення, отримання макетних зразків що реалізують цю систему виконано такі завдання:

- розроблено алгоритми кодування текстових повідомлень для реалізації з використанням обчислювальної системи (алгоритм містить процедури перетворення текстової інформації в десяткове ціле число, представлення цього числа у факторіальній системі числення з подальшим перетворенням в перестановку);
- розроблено алгоритм декодування текстових повідомлень для реалізації з використанням обчислювальної системи (алгоритм містить процедури перевірки перестановки на коректність і належність до дозволеної множини перестановок, а також послідовне перетворення перестановки в число в факторіальній, десятковій і двійковій системах числення і, насамкінець, представлення двійкового числа в символьному вигляді);
- розроблено структурну схему, програмний код і макетні зразки системи передавання даних.

4.3.1. Алгоритм кодування текстових повідомлень в перестановку

Алгоритм перетворення передбачає перетворення текстових повідомлень у ФКВД [7], за довжиною перестановки $M = 8$ потужність повної множини алфавіту становить $M! = 40320$. Цієї кількості перестановок

достатньо для кодування повідомлення довжиною до $n = \lfloor \log_2 M! \rfloor = \lfloor \log_2 40320 \rfloor = 15$ біт.

Опис алгоритму обробки тексту:

1. Повідомлення, введені користувачем, розбиваються на групи по два символи (якщо кількість символів у повідомленні непарна, останнім символом додається символ пробілу).
2. З отриманих пар символів формується числове значення наступним чином. Від ASCII коду кожного символу беруться молодші 7 біт (цієї кількості достатньо для передавання великих і малих латинських символів, чисел, деяких додаткових символів) і конкатенуються у 14-бітове двійкове число. Старшими 7 бітами є ASCII код першого символу, молодшими 7 бітами – ASCII код другого символу).
3. Отримане 14-бітове двійкове число переводиться в десяткову систему числення. Це число однозначно ідентифікує дві літери та може бути перетворене в текст на приймальній стороні шляхом зворотного переведення у двійкову систему числення та розбиття отриманого значення на два 7-розрядних двійкових числа з подальшою інтерпретацією їх як ASCII кодів символів текстового повідомлення користувача.
4. Після перетворення двох текстових символів у десяткове число воно представляється у факторіальній системі числення $x = \sum_{k=1}^n d_k \times k!$, де $0 \leq d_k \leq k$.
5. З отриманого факторіального числа на основі деякої базової перестановки π_0 формується перестановка π . Процедuru такого перетворення детально описано в [8].

Я вже було зазначено використання перестановок довжиною $M = 8$ має потужність алфавіту більшу ($8! = 40320$) за загально можливу кількість

двохсимвольних комбінацій ($2^{14} = 16384$). Це дозволяє сформувати ансамбль використовуваних перестановок і вести контроль за належністю прийнятої перестановки до цього ансамблю, а також відкидати її в разі належності до множини перестановок, яка не використовується джерелом.

4.3.2. Приклад перетворення послідовності текстових символів у перестановку

Для перетворення двох латинських символів ab у перестановку визначаються ASCII коди цих символів – 97_{10} та 98_{10} відповідно. Далі виконується переведення отриманих значень у двійкову систему числення та конкатенацію молодших 7 біт чисел. У результаті отримане 14-бітне значення 11000011100010_2 еквівалентне 12514_{10} . Отримане число 12514_{10} представляється у факторіальній системі числення:

$$\begin{aligned} 12514_{10} &= 2 \times 7! + 3 \times 6! + 2 \times 5! + 1 \times 4! + \\ &+ 1 \times 3! + 2 \times 2! + 0 \times 1! + 0 \times 0! = 23211200_F \end{aligned}$$

Для формування перестановки обирається базова перестановка (ця перестановка тримається в секреті і не розголошується учасниками інформаційного обміну) $\pi_0 = (0, 1, 2, 3, 4, 5, 6, 7)$. Кожний розряд отриманого числа у факторіальній системі числення є індексом елемента базової перестановки. Почергово аналізуючи розряди факторіального числа, зчитуються відповідні елементи базової перестановки. При цьому зчитаний елемент вилучається з неї. Для числа 23211200_F старший розряд, а отже й індекс у базовій перестановці, дорівнює 2. Другий елемент базової перестановки π_0 (число 2) зчитується, записується першим елементом результуючої перестановки π і вилучається з базової перестановки π_0 . На поточному етапі базова перестановка π_0 буде мати наступний вигляд: $\pi'_0 = (0, 1, 3, 4, 5, 6, 7)$. Наступний розряд факторіального числа 23211200_F (число 3). За елементом з індексом 3 з перестановки π'_0 буде число 4, що буде

другим елементом перестановки π й видалимо значення 4 з базової перестановки.

Повторивши описаний алгоритм перетворення факторіального числа в перестановку для всіх розрядів числа 23211200_F , отримаємо перестановку $\pi = (2, 4, 3, 1, 5, 7, 0, 6)$. Таким чином, текстову послідовність із двох символів ab можна представити перестановкою $(2, 4, 3, 1, 5, 7, 0, 6)$. Таку перестановку можна передати каналом зв'язку та однозначно перетворити в послідовність символів на приймальній стороні.

При передаванні перестановок радіоканалом не обов'язково представляти елементи перестановки у вигляді ASCII символів, що займають 1 байт. При зазначеній довжині перестановки $M = 8$, елементи будуть лежати в проміжку від 0 до 7. Для кодування 8 значень у двійковій системі числення достатньо $\log_2 8 = 3$ біти. Тому, виконавши конкатенацію трьох молодших біт кожного з елементів перестановки, можна упакувати двійкове представлення перестановки в $3 \times 8 = 24$ біти або 3 байти, передавання яких і виконується каналом зв'язку.

При інформаційному обміні відправник і отримувач повинні використовувати одну й ту ж базову перестановку π_0 . Тільки в цьому випадку повідомлення може бути коректно декодовано. Якщо відправник і отримувач будуть зберігати базову перестановку π_0 , яка сформована випадковим чином, у секреті, канал передавання буде захищеним. Для такого випадку виникає задача узгодження початкової перестановки між абонентами. Цю задачу можливо вирішити, наприклад, за допомогою використання трьохетапного криптографічного протоколу на основі перестановок [9].

4.3.3. Алгоритм приймання та декодування перестановок у текст

Під час отримання з каналу зв'язку перестановки приймач може визначити, чи перестановку прийнято коректно. Для цього перевіряються всі отримані $M = 8$ елементів (24 біти). Кожний елемент із множини цілих

значень діапазону $[0, M-1]$ повинен бути присутнім і траплятися у перестановці лише один раз. Якщо в прийнятій послідовності є декілька однакових елементів або деякі елементи відсутні, таку послідовність не можна вважати перестановкою. Вона може бути відкинута, а передавач змушений буде виконати повторне передавання перестановки. Крім того, оскільки відповідно до [7] параметри ФКВД обираються таким чином, щоб виконувалася рівність $M! \geq 2^k$, де k – кількість біт, що необхідно передати, завжди будуть присутні $M! - 2^k$ перестановок, які не використовуються джерелом. Перестановки з цієї множини теж необхідно обробляти як некоректні.

Алгоритм перетворення отриманих перестановок полягає в наступному:

1. Приймач отримує перестановку, яку необхідно перетворити на послідовність текстових символів.
2. Отриману перестановку перетворюють в число у факторіальній системі числення. Процедура перетворення аналогічна перетворенню числа у факторіальній системі числення в перестановку за допомогою базової перестановки π_0 , що відбувається на етапі відправки повідомлення. (Кожний елемент перестановки вказує на індекс елемента базової перестановки π_0 , вилучаючи при цьому елемент з базової перестановки).
3. Факторіальне число перетворюється в десяткове.
4. Десяткове число подають в двійковому вигляді і розділяють 14-ти бітне число на два 7-ми бітних числа ASCII символів, що були передані передавачем.

4.3.4. Приклад декодування перестановки у текст

Перетворення відбувається наступним чином. Нехай отримано перестановку $\pi = (2, 4, 6, 3, 0, 7, 1, 5)$. Спершу отриману перестановку перетворюють в число у факторіальній системі числення. Базова перестановка

при цьому становить $\pi_0 = (0, 1, 2, 3, 4, 5, 6, 7)$, старший елемент перестановки π (число 2) визначає індекс (число 2) старшим розрядом у факторіальному числі, вилучаючи елемент зі значенням 2 з базової перестановки π_0 . На поточному етапі початкова перестановка π_0 перетворюється в $\pi'_0 = (0, 1, 3, 4, 5, 6, 7)$. Наступний елемент перестановки (число 4) визначає індекс (число 3) наступним розрядом у факторіальному числі, вилучаючи елемент зі значенням 4 з перестановки π'_0 . Повторивши алгоритм для всіх елементів перестановки π , формується число 23420200_F . Після його перетворення в десяткове число отримаємо:

$$23420200_F = 2 \times 7! + 3 \times 6! + 4 \times 5! + 2 \times 4! + \\ + 0 \times 3! + 2 \times 2! + 0 \times 1! + 0 \times 0! = 12772_{10}$$

Десяткове число 12772_{10} дорівнює двійковому 11000111100100_2 , яке розділяється на дві групи по 7 біт: 1100011_2 , 1100100_2 . При перетворенні отриманих значень в символи за стандартною таблицею ASCII символів формуються текстовий еквівалент символів *cd*.

4.3.5. Структура і опис прототипу системи захищеного інформаційного обміну текстовими повідомленнями ISM радіоканалом

ISM (Industrial, Scientific, and Medical) радіоканал — це діапазон радіочастот, призначений для неліцензованого використання в промислових, наукових і медичних застосуваннях. Основними діапазонами ISM є 433 МГц, 868 МГц, 915 МГц і 2,4 ГГц, які широко застосовуються в бездротових мережах, IoT-пристроях, телеметрії та RFID-системах. Особливістю ISM-каналу є його стійкість до перешкод за рахунок методів розширеного спектра (FHSS, DSSS), що забезпечує надійну передачу даних. Висока доступність і глобальне поширення роблять ISM-діапазони ключовими для розвитку бездротових технологій. Відповідно пристрій передачі та прийому складових інформаційного обміну застосовано система на кристалі (System on Chip –

SoC) nRF52840 компанії Nordic Semiconductor [10]. Пристрій nRF52840 має вбудоване ядро ARM Cortex-M4, що працює на частотах до 64 МГц, радіомодуль на частоті 2,4 ГГц, що підтримує різноманітні протоколи передавання (Bluetooth LE, Bluetooth mesh, Thread, Zigbee, 802.15.4, ANT, ESB), 1 MB Flash пам'яті, 256 KB оперативної пам'яті, вбудований апаратний USB 2.0 контролер з можливістю роботи в режимі CDC (USB serial).

На (Рисунок 4.24) та (Рисунок 4.25) наведемо структурні схеми систем передавання інформації на основі ФКВД.

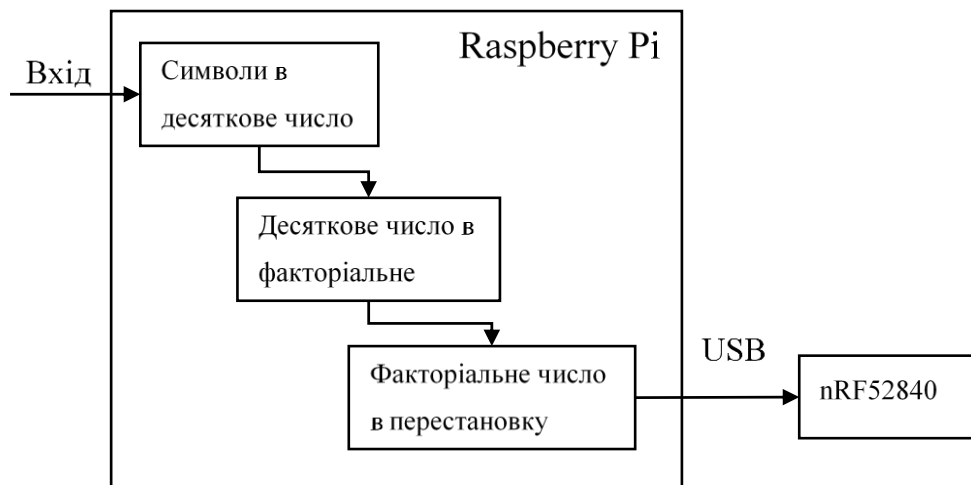


Рисунок 4.24 – Структурна схема системи передавання

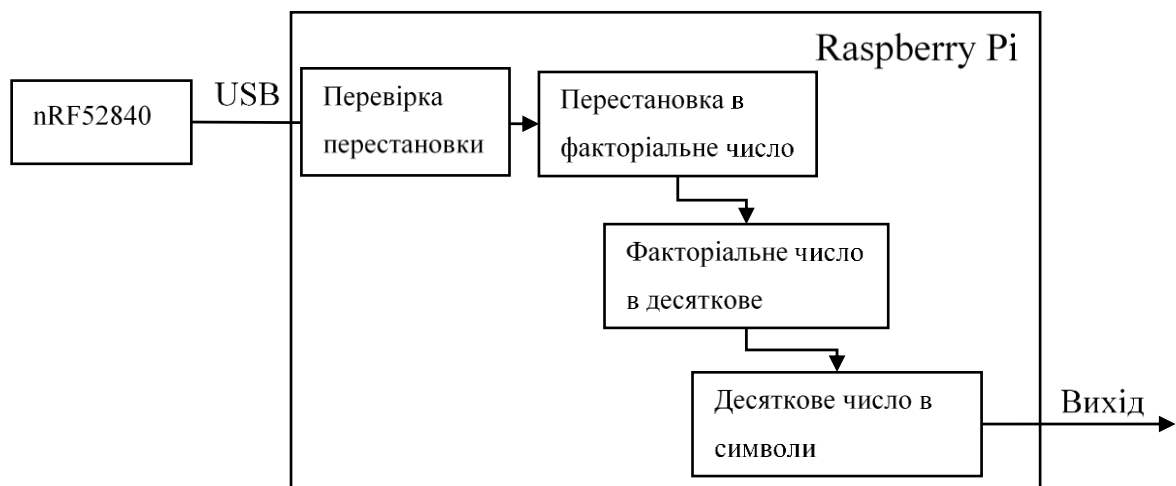


Рисунок 4.25 – Структурна схема системи приймача

Передавання інформації відбувається в пакетному режимі з фіксованою довжиною пакета, що дорівнює 3 байтам (прийнятої для $M = 8$).

Використання контролера nRF52840 дозволяє вести передавання інформації з будь-якого пристрою, що підтримує інтерфейс USB, та на якому запущено відповідне програмне забезпечення.

Варто зазначити, що протокол передавання даних не прив'язаний до апаратного забезпечення і може використовувати практично будь-які приймально-передавальні радіопристрої в різних діапазонах частот. З метою передавання інформації на частотах 433 МГц або 868 МГц (ISM-діапазони) може бути використаний контролер серії CC1352P компанії Texas Instruments [11]. ISM (industrial, scientific and medical) - діапазони доступні для безліцензійного використання практично в будь-якій країні. В Україні список доступних частот регламентується Постановою Кабінету Міністрів України № 1208 від 15.12.2005 р. «Про затвердження Національної таблиці розподілу смуг радіочастот України» [12].

В якості комп'ютерної системи що генерує та перетворює інформаційні повідомлення в ФКВД використано плату Raspberry Pi 4 Model B [13], на якій запущено програмне забезпечення створено з використанням мови програмування Python [14]. Raspberry Pi 4 Model B – це одноплатний комп'ютер на основі ARM v8 архітектури, що працює з використанням операційної системи Linux. Ітерфейсом підключення nRF52840 та Raspberry Pi обрано інтерфейс USB, а для самого радіомодуля розроблено програмне забезпечення, що дозволяє виконувати двосторонній обмін перестановками радіоканалом через віртуальний COM-порт.

Фото макетних зразків приймально-передавальних пристроїв системи захищеного інформаційного обміну на основі ФКВД показано на (Рисунок 4.26).

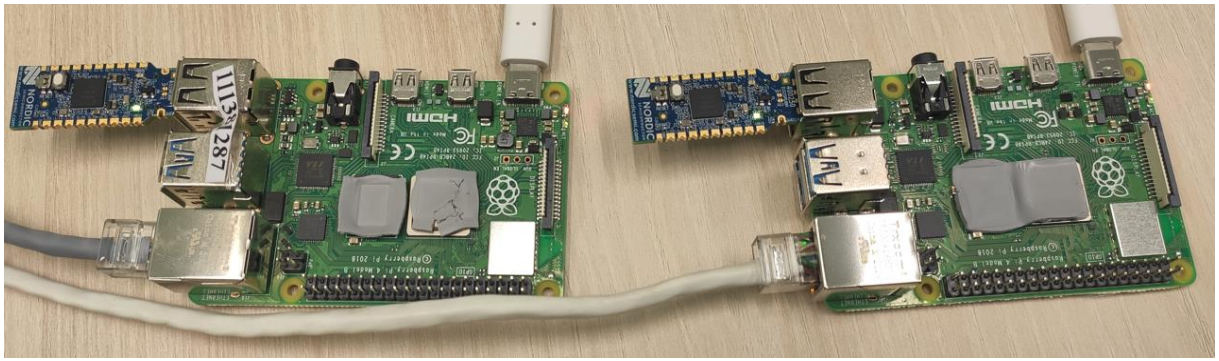


Рисунок 4.26 – Макетні зразки системи захищеного інформаційного обміну

Перевірка функціонування створених макетних зразків на базі одноплатного комп'ютера передбачає використання командного рядок віддаленого SSH доступу до Raspberry Pi (Рисунок 4.27) та (Рисунок 4.28).

 A terminal window titled 'pi@raspberrypi: ~' is shown. The command 'sudo python3 main.py -c /dev/ttyACM0 -m 'test message'' has been entered and executed. The output consists of several lines of hexadecimal data: 27536140, 27450613, 06712543, 25063741, 27431560, 25364071, and 03276145. The prompt 'pi@raspberrypi:~ \$' is visible at the bottom.

Рисунок 4.27 – Процес передавання повідомлення

 A terminal window titled 'pi@raspberrypi: ~' is shown. The command 'sudo python3 main.py -c /dev/ttyACM0 -p' has been entered and executed. The output is 'test message'. The prompt 'pi@raspberrypi:~ \$' is visible at the bottom.

Рисунок 4.28 – Процес прийому повідомлення

Для початку отримання повідомлень на приймальній стороні необхідно запустити програму з ключем командного рядка *-p*. Для передавання повідомлення на передавальній стороні необхідно запустити програму з ключем командного рядка *-m* і ввести повідомлення, яке необхідно передати. В обох випадках додатково необхідно вказати інтерфейс віртуального СОМ порту за допомогою ключа командного рядка *-c*, через який буде вестись передавання. Під час передавання повідомлення в консоль передавача виводяться перестановки, що надсилаються в канал зв'язку. Приклад

реалізації процесу передавання та приймання повідомлення з використанням створених макетних зразків наведено на (Рисунок 4.27) та (Рисунок 4.28). Приймальна сторона системи очікує нового повідомлення, виконує перевірку перестановки, перетворює її на текст та виводить користувачу

4.3.6. Особливості макетних зразків

Запропоновану реалізацію системи захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону можна використати для обміну як символьними, так і двійковими даними різної довжини. За рахунок збереження базової перестановки в секреті досягається конфіденційність зв'язку між абонентами. З метою узгодження ключової базової перестановки передбачається можливість застосування трьохетапного криптографічного протоколу [181].

Побудовану інформаційну систему та розроблені макетні зразки приймально-передавальних пристроїв можна використати як базові елементи для подальшої реалізації більш складних систем на основі ФКВД. В поєднанні з методами надійного передавання перестановок та організації надійної синхронізації при передаванні через канал з високою інтенсивністю шуму існує перспектива створення достатньо надійної та захищеної системи інформаційного обміну за ймовірності бітової помилки, близької до 0.5.

Проведені експерименти мали умову високих показників відношення сигнал/шум у каналі зв'язку, що вносить нехтовно малі значення відносної частоти бітової помилки ($\sim 10^{-5}$). Разом з тим, у подальшому дослідженні макетних зразків необхідно виконати аналіз достовірності передавання інформації в умовах збільшення інтенсивності появи помилок – за умови збільшення відстані між передавачем і приймачем, зменшення потужності передавача, чутливості приймача чи створення штучних шумів у каналі.

Крім того, розроблена система не повною мірою використовує властивості та переваги ФКВД. Для виправлення цієї ситуації потрібно

розробити каналний протокол, де б кодові слова ФКВД відігравали роль кадрів.

4.4. Використання графічних прискорювачів для операцій над перестановками

Згаданий трьохетапний криптографічний протокол обміну ключовими перестановками є математично складним у певних операціях при використанні ФКВД. Такі особливості як довжина перестановки, кількість блоків синхронізації, кількість наступних інформаційних фрагментів накладають певні обмеження при розробці власного протоколу на основі ФКВД. Однак при розгляді сучасних рішень для світу IoT певні обмеження можна обійти, або зменшити їх вплив шляхом використання технологічних особливостей тієї чи іншої обчислювальної платформи та її апаратних засобів.

Метою розглянутого в цьому підрозділі дослідження є збільшення швидкості виконання операцій над перестановками, які використовуються в трьохетапному криптографічному протоколі [181] за рахунок використання платформи CUDA (Compute Unified Device Architecture).

Цілями цього дослідження є:

- визначення найбільш часто використовуваних операцій над перестановками.
- створення програмного коду, який реалізує операції над перестановками і може використовувати апаратне забезпечення графічних процесорів.
- отримання оцінки продуктивності різних підходів, включаючи обчислення на CPU та різні режими обчислень на GPU.

На 2024 рік існує здебільшого дві окремі компанії Nvidia та AMD, які виробляють високопродуктивні графічні процесори. Такі процесори можна використовувати для обчислень загального призначення (GPGPU). Графічні процесори компаній мають платформи, Compute Unified Device Architecture (CUDA) від Nvidia і Advanced Micro Devices (ROCm) від AMD. Ці платформи

дозволяють розробникам легко використовувати своє обладнання для GPGPU. У подальшому розгляді використано графічний процесор виробника Nvidia CUDA, з метою збільшення швидкості виконання операцій над перестановками за одиницю часу.

Платформа Nvidia CUDA дозволяє графічним процесорам паралельно виконувати обчислення, використовуючи сотні й тисячі ядер GPU. Такий підхід значно прискорює роботу різних алгоритмів, які обробляють великі блоки даних. Такими процесами можуть бути сортування, вейвлет-перетворення (операції над сигналами та зображеннями), криптографія, молекулярна динаміка, машинне навчання тощо. Починаючи з графічного процесора Nvidia GeForce 8800, усі подальші версії графічних процесорів Nvidia підтримують CUDA API [182]. CUDA можна використовувати з Microsoft Windows і різними дистрибутивами Linux (Debian, CentOS, Ubuntu тощо). Розробники мають різноманітний інструментарій під різні мови програмування C, C++ і Fortran. Крім того, доступні сторонні обгортки для інших популярних мов програмування, таких як Python або Java.

4.4.1. Особливості архітектури графічних процесорів

Перед використанням ядер CUDA нагадаємо архітектуру графічного процесора. Процесор GA102 побудований на основі сучасної архітектури Nvidia Ampere [19], основні принципи його організації зберігаються і в попередніх поколіннях GPU (Graphics Processing Unit).

Типовий графічний процесор Nvidia складається з кластерів обробки графіки (GPC – Graphics Processing Cluster), кластерів обробки текстур (TPC – Texture Processing Cluster), мультипроцесорів потокової обробки (SM – Streaming Multiprocessor), контролерів пам'яті та інших функціональних блоків. У чіпі GA102 є 7 GPC, кожен з яких містить 6 TPC, а кожен TPC – 2 SM.

Ядра CUDA є складовою частиною SM, і кожен мультипроцесор містить 128 ядер CUDA, згрупованих у 4 секції по 32 ядра. Загальна кількість ядер

CUDA у чіпі GA102 становить 10 752, що дозволяє ефективно використовувати його для загальних обчислень на графічному процесорі (GPGPU – General-Purpose Computing on Graphics Processing Units).

Кожне ядро CUDA має власні обчислювальні блоки та регістри, які дозволяють виконувати операції як з цілими числами, так і з числами з плаваючою комою. Ядра мають доступ до пам'яті пристрою, що дає змогу швидко завантажувати дані в регістри, виконувати обчислення та передавати результати назад.

Висока продуктивність забезпечується ефективною пам'яттю графічного процесора, зокрема GDDR6X (Graphics Double Data Rate 6X) або HBM2 (High Bandwidth Memory 2), які мають пропускну здатність до 2039 ГБ/с. Додатково чіп містить вбудовані кеші, що прискорюють обмін даними.

Мікросхема GA102, як і більшість сучасних графічних процесорів із підтримкою CUDA, використовує кілька типів апаратної пам'яті:

- Регістри – 16 384 32-розрядні загальні регістри для кожного розділу в SM.
- Кеш інструкцій L0 – один для кожного розділу.
- Кеш даних L1 – 128 КБ (по одному для кожного SM).
- Кеш L2 – 6 144 КБ (спільний для всіх SM).
- Пам'ять пристрою – GDDR6X або HBM2.

Модель програмування CUDA надає розробникам доступ до цих типів пам'яті у такій формі:

- Регістри – апаратні регістри, доступні кожному потоку.
- Спільна пам'ять – частина кешу L1, яку можна налаштовувати залежно від завдання.
- Локальна пам'ять – фізично розташована в пам'яті пристрою, але використовується як приватна пам'ять для окремих потоків.
- Глобальна пам'ять – основна пам'ять пристрою, доступна всім потокам.
- Постійна пам'ять – також розташована в пам'яті пристрою, але кешується для швидкого доступу.

4.4.2. Застосування графічних процесорів у криптографічних перетвореннях

Відомі різні підходи, які можуть використовувати описане обладнання GPU для збільшення швидкості криптографічних протоколів. У дослідженні [183] автори отримали 20-кратне збільшення алгоритму передового стандарту шифрування (AES) з використанням платформи CUDA. У дослідженні [184], автори реалізували високопродуктивний сервер підписів, який реалізує алгоритм цифрового підпису еліптичної кривої (ECDSA) з використанням прискорення GPU. У дослідженні [185], показані загальні методи реалізації та прискорення різних блокових шифрів, доступних у криптографічному механізмі OpenSSL, за допомогою GPU.

Трьохетапний криптографічний протокол на основі перестановок [181] використовує визначені математичні операції над перестановками, відповідно виконання цих операцій можна прискорити за допомогою інструкцій SIMD. Однією з найбільш використовуваних операцій над перестановками в трьохетапному протоколі є множення перестановок. Ця операція використовується для обчислення значень $Y_1 - Y_4$, ключів σ_A і σ_B , щоб визначити повідомлення π .

Далі аналізується випадок передавання повідомлення від Аліси до Боба із застосуванням протоколу [181] У цьому випадку Аліса і Боб утворюють перестановку α і обчислюють її розкладання в добуток непересічних циклів, отримані значення її загальновідомими.

Аліса генерує свій секретний ключ \bar{s} , ключову перестановку σ_A та її зворотний σ_A^{-1} .

Боб генерує свій секретний ключ \bar{r} , перестановку ключа σ_B , та її зворотний σ_A^{-1} . Крім того, Боб генерує перестановку χ_B . Аліса та Боб зберігають ці дані окремо один від одного та не поширюють.

Аліса відображає повідомлення яке необхідно передати m на перестановку π , обчислює $Y_1 = \sigma_A \cdot \pi$ і надсилає Бобу.

Боб отримує Y_1 , обчислює $Y_2 = \sigma_B \cdot Y_1 \cdot \chi_B$ і відправляє його до Аліси.

У свою чергу Аліса обчислює π^{-1} та значення $Y_3 = \sigma_A^{-1} \cdot Y_2 \cdot \pi^{-1}$ і надсилає Y_3 назад Бобу.

Боб обчислює значення $Y_4 = \sigma_B^{-1} \cdot Y_3$ і перетворює його на добуток непересічних циклів. Подальший порядок дії спрямований на отримання початкової перестановки π , яку можна знайти шляхом обчислення групи множень $Y_1 \cdot \pi'^{-1}$

де π' припущена перестановка, яка формується залежно від Y_4 непересічних циклів. Після цього початкове повідомлення t може бути відновлено.

Загалом близько половини операцій над перестановками, що виконуються в протоколі, є операціями множення. Добутком перестановок A і B довжини M є перестановка $C = A \cdot B$, де $C[i] = A[B[i]]$, $A[i]$, $B[i]$, $C[i]$ — елементи перестановок у позиції i , $i \in [0; M - 1]$. Отже, множення перестановок - це перетворення одного значення в пам'яті в інше значення в пам'яті. Ця операція є швидкою та незалежною (кожен елемент множення перестановки може бути обчислений незалежно від інших елементів), тому ідеально виконувати її паралельно. Графічні процесори Nvidia мають тисячі ядер CUDA (у порівнянні з сучасними процесорами, які мають десятки ядер) із швидким доступом до пам'яті, як описано раніше, і можуть виконувати цю операцію надзвичайно швидко.

4.4.3. Оцінка швидкодії виконання операцій над перестановками при використанні графічних процесорів

Очікується, що оптимізація множення перестановок призведе до максимального збільшення швидкодії. Підвищення продуктивності за допомогою сучасних графічних процесорів оцінюється в рамках цього дослідження.

CUDA використовується разом із набором для розробки програмного забезпечення, який дозволяє застосовувати мову програмування C++ для

створення програм. Для порівняння швидкодії операцій множення над перестановками розроблено алгоритм множення двох перестановок із використанням різних апаратних платформ: CPU (без інструкцій SIMD) і GPU (із застосуванням CUDA). Для порівняння продуктивності CPU та GPU використано бібліотеку Google Benchmark [186]. В процесі порівняння довжина перестановок є фіксована та становить 32 768 елементів. Результати випробувань наведено в (Таблиця 4.1).

Таблиця 4.1 – Порівняння продуктивності однієї операції множення двох перестановок

Обладнання	Множення в секунду
CPU (Ryzen 5 3600)	12466
GPU (GTX 1070)	3174
GPU (GTX 1650 GDDR6)	2756

У (Таблиця 4.1) продемонстровано, що продуктивність множення перестановок на GPU значно нижча порівняно з CPU. Це суперечить очікуванням, оскільки графічний процесор зазвичай забезпечує значно вищу продуктивність у подібних обчислювальних задачах.

Цей результат пояснюється особливостями архітектури GPU. Графічний процесор є дискретним обчислювальним пристроєм із власними обчислювальними блоками та пам'яттю. Він взаємодіє з центральним процесором (CPU, хостом) через шину даних, що забезпечує обмін інформацією. Оскільки GPU не може використовуватися для загальних операцій введення-виведення, усі необхідні дані спочатку обробляються CPU і його пам'яттю, після чого передаються в пам'ять GPU, обробляються там, а результати повертаються назад у пам'ять CPU.

Основним фактором, що знижує ефективність множення перестановок на GPU, є значні витрати часу на передавання даних між пам'яттю CPU та GPU. Навіть якщо сам процес множення перестановок займає мінімальний час,

затримки, пов'язані з копіюванням даних, значно впливають на загальну продуктивність.

Щоб кількісно оцінити ці витрати, було використано інструмент профілювання *nvprof* від Nvidia, який дозволяє визначити час, витрачений на різні етапи обчислень.

Вимірювання показали, що:

- близько 55% загального часу витрачається на копіювання даних із пам'яті CPU в пам'ять GPU;
- приблизно 30% часу йде на повернення результатів із пам'яті GPU в пам'ять CPU;
- лише 15% часу витрачається безпосередньо на виконання множення перестановок.

Для експериментального дослідження продуктивності множення великої кількості перестановок було згенеровано фіксовану кількість перестановок у пам'яті хоста та негайно завантажено їх у пам'ять GPU. Далі виконувалося багаторазове множення перестановок, причому всі проміжні результати зберігалися в пам'яті GPU. Після завершення обчислень результати поверталися у пам'ять CPU для подальшої обробки.

Для експерименту використовувався графічний процесор Nvidia GTX 1070, який побудований на архітектурі Pascal, має 1920 ядер CUDA та 8 ГБ пам'яті GDDR5. Результати випробувань наведено у (Таблиця 4.2).

Таблиця 4.2 – Продуктивність багаторазового множення перестановок

Кількість множень в операції	Операцій в секунду	Еквівалентна кількість разових множень за секунду
1	3174	3174
2	3174	6348
5	2987	14935
10	2869	28690
20	2354	47080
50	1706	85300

100	1220	122000
200	746	149200
500	346	173000
1000	188	188000
2000	93	186000
3000	62	186000

У цьому дослідженні для оцінки продуктивності введено поняття "операція", яке означає множення групи перестановок. Наприклад, множення 100 перестановок одночасно вважається однією операцією. Еквівалентну кількість одиничних множень перестановок можна визначити, помноживши загальну кількість множень перестановок на кількість виконаних операцій за секунду.

4.4.4. Оцінка результатів використання графічних процесорів для операцій над перестановками

Паралельне множення перестановок на GPU можливо використовувати в таких завданнях, як атаки грубої сили. Кryptoаналітику потрібно завантажити операнди (пари перестановок, які потрібно перемножити) у пам'ять GPU, виконати кілька множень паралельно та отримати всі результати, щоб перевірити, чи знайдено ключ протоколу.

Експерименти показують, що продуктивність GPU залежить від кількості множень перестановок, зроблених за одну операцію. Результати в (Таблиця 4.2) демонструють, що множення набору перестановок відбувається набагато швидше, коли кількість перестановок збільшується. Таким чином продуктивність GPU перевищує продуктивність CPU, починаючи з 5 перестановок одночасно. В роботі [187] показано можливість підвищити продуктивність множення перестановок процесора за допомогою інструкцій SIMD приблизно в 2,5 рази. Однак використовувати GPU буде набагато ефективніше для виконання операцій множення.

У випадку атаки грубої сили криптоаналітик володіючи алгоритмом трьохетапного протоколу [181] знає що секретні ключі \bar{s} та \bar{r} мають вигляд вектора:

$$\bar{s} = (s_1, s_2, \dots, s_{n(\alpha)}), \quad (4.11)$$

де $0 \leq s_i \leq l(\alpha_i) - 1$, $l(\alpha_i)$ - порядок циклу α_i при розкладанні на добуток непересічних циклів $\alpha = \prod_{i=1}^{n(\alpha)} \alpha_i$.

Значення \bar{s} і \bar{r} використовуються для створення ключових перестановок $\sigma_A = \prod_{i=1}^{n(\alpha)} \alpha_i^{s_i}$ і $\sigma_B = \prod_{i=1}^{n(\alpha)} \alpha_i^{r_i}$. Таким чином, якщо криптоаналітик виконує атаку грубої сили та знайти перестановку ключа, він повинен пройти через усі можливі значення s_i та обчислити значення σ для кожного набору вектора (4.11). При цьому кількість можливих векторів складає $\alpha = \prod_{i=1}^{n(\alpha)} l(\alpha_i)$.

У роботі довжина перестановки прийнята такою що дорівнює довжині сучасних шифрів і становить 128, 192 або 256 біт. Для здійснення атаки грубої сили, криптоаналітик перебирає всі можливі ключі, кількість яких дорівнює 2^k , де k є довжиною ключа. Для знаходження еквівалентної структури α криптоаналітик виконує порівняння кількості можливих значень ключа з мінімальною можливою довжиною перестановки.

При $l(\alpha_i) = l(\alpha_j)$, для всіх i, j кількість можливих значень ключа σ становитиме $l(\alpha)^{n(\alpha)}$. Довжина ключа σ , що дорівнюватиме довжині ключа сучасних шифрів, повинна відповідати нерівності $l(\alpha)^{n(\alpha)} \geq 2^k$. Іншими словами, криптоаналітику потрібно знайти $n(\alpha) = \lceil \log_{l(\alpha)} 2^k \rceil$ для кожної довжини ключа $k \in \{128, 192, 256\}$. Результати розрахованих значень наведені в (Таблиця 4.3).

Таблиця 4.3 – Еквівалентні α -структури за різних довжин ключа

Довжина альфа-циклу	Число альфа-циклів	Довжина альфа- перестановки	Довжина альфа-циклів	Число альфа-циклів	Довжина альфа- перестановки	Довжина альфа-циклів	Число альфа-циклів	Довжина альфа- перестановки
2^{128}			2^{192}			2^{256}		
3	81	243	3	122	366	3	162	486
4	64	256	4	96	384	4	128	512
5	56	280	5	83	415	5	111	555
6	50	300	6	75	450	6	100	600
7	46	322	7	69	483	7	92	644
8	43	344	8	64	512	8	86	688
9	41	369	9	61	549	9	81	729
10	39	390	10	58	580	10	78	780

Мінімально прийнятна довжина α становить 243 елементи для довжини ключа $k=128$, 366 елементів для довжини ключа $k=192$ і 486 елементів для довжини ключа $k=256$ при $l(\alpha_i) = 3$.

У випадку $l(\alpha_i) = 3$, $n_\alpha = 81$ щоб виконати атаку грубої сили, криптоаналітик повинен створити 3^{81} вектор \bar{s} . Для обчислення вектору \bar{s} і знаходженню значення σ_A , необхідно виконати мінімальну кількість множень перестановок із 81. Отже, кількість перестановок, необхідних для знаходження ключової перестановки σ_A дорівнює $81 \cdot 3^{81}$.

За результатами наведеної продуктивності в таблиці 4.2, знадобиться до $\frac{81 \cdot 3^{81}}{188000} \approx 1.91 \cdot 10^{35}$ секунд, щоб криптоаналітик знайшов ключ σ_A (при використанні графічного процесора GTX1070), що є значним значенням.

Результати експериментів демонструють, що продуктивність множення перестановок на GPU починає перевищувати продуктивність аналогічної операції на CPU, коли кількість паралельних множень досягає п'яти. За цих умов швидкість обчислень складає приблизно 14 900 множень на секунду. Максимальна продуктивність спостерігається при 1000 паралельних множеннях, досягаючи значення близько 188 000 множень на секунду.

Продуктивність продовжує зростати, доки не буде досягнуто оптимального співвідношення між передаванням даних у пам'ять GPU, їх поверненням та виконанням множення.

Застосування атаки методом грубої сили до криптографічних ключів на основі перестановок потребуватиме значного часу, за умови правильного вибору параметрів перетворення трьохетапного криптографічного протоколу.

Ще одним можливим застосуванням графічного процесора є використання його як центрального серверного обчислювального вузла у центрі обробки даних, здатного обслуговувати велику кількість клієнтів із використанням криптографічного протоколу. Запити від різних клієнтів можуть бути згруповані та передані на GPU для виконання множення перестановок. У такій конфігурації всі операції передавання даних між пам'яттю та обчислення множень перестановок можуть виконуватися одночасно, що сприяє досягненню максимальної продуктивності.

Як видно з (Таблиця 4.1), використання GPU для одиничного множення перестановок не забезпечує збільшення продуктивності порівняно з CPU, що пояснюється особливостями архітектури графічного процесора. Водночас, підвищення швидкодії можливе за умови одночасного виконання великої кількості множень перестановок, як за результатами наведеними в таблиці 4.2.

Таблиця 4.3 містить відповідні значення довжини α , що дозволяють припустити (на основі очікуваного часу обчислень), що знаходження перестановки ключа для трьохетапного криптографічного протоколу на основі перестановок є вкрай складним завданням при $\alpha > 243$ елементи, навіть за використання спеціалізованого обчислювального обладнання. Запропонований підхід може бути застосований і в інших сферах, зокрема у задачах, пов'язаних із факторіальним кодуванням [188], [189], [190], [191]

Варто зазначити, що криптоаналітик може використати більш потужний графічний процесор для здійснення атаки. Продуктивність множення перестановок можна ще покращити шляхом використання сучасних графічних процесорів серії Nvidia RTX, які мають до 10752 ядер CUDA та швидко

пам'ять GDDR6X із підвищеною пропускною здатністю. Крім того, можлива паралельна робота декількох GPU, що потенційно може збільшити загальну продуктивність на порядок. Однак, навіть за таких умов час, необхідний для перебору ключів, залишається критичним фактором.

Ще одним підходом для подальшого підвищення продуктивності трьохетапного криптографічного протоколу на основі перестановок є використання програмованих вентильних матриць (FPGA – Field-Programmable Gate Array). Технологія FPGA дозволяє створювати спеціалізовані апаратні блоки, що прискорюють виконання інтенсивних обчислювальних задач та дають змогу реалізувати криптографічний **протокол** повністю на апаратному рівні. Різні підходи до реалізації криптографічних протоколів за допомогою FPGA детально описані в літературі [192], [193], [194], [195], [196].

4.5. Висновки

У цьому розділі наведено практичні аспекти реалізації систем інформаційного обміну на основі перестановок. Основні результати роботи зосереджено на моделюванні та експериментальній перевірці ефективності НФКД при реалізації інформаційного обміну на основі перестановок, а також дослідженні можливостей апаратних прискорювачів для використання у криптографічних операціях з використанням перестановок.

Розроблено імітаційну модель системи інформаційного обміну з нероздільним факторіальним кодуванням даних. За допомогою розробленої імітаційної моделі оцінено вплив помилок на достовірність отриманих перестановок, визначено частоту приймання хибних перестановок. Проведено серію експериментів із передавання перестановок двійковим симетричним каналом, що підтвердили адекватність розробленої моделі.

Створено макетні зразки приймально-передавальних пристроїв захищеного інформаційного обміну текстовими повідомленнями на основі перестановок ISM-радіоканалом. У основу покладено пристрій nRF52840, що

забезпечує ефективне бездротове передавання перестановок. Виконано тестування макетних зразків, яке підтвердило працездатність запропонованої системи та можливість її використання для надійного обміну текстовими повідомленнями.

Розроблено алгоритм кодування текстових повідомлень у перестановку на основі факторіальної системи числення. Розроблено алгоритм декодування перестановки в текстове повідомлення.

Підтверджено ефективність використання графічних процесорів для прискорення криптографічних операцій над перестановками. Визначено, що ефективність множення перестановок на GPU перевищує продуктивність CPU за паралельного виконання великої кількості операцій. Водночас, для одиничних множень перевага залишається за CPU через значні витрати часу на передавання даних між пам'яттю CPU та GPU.

Результати цього розділу мають значну практичну цінність. Розроблена імітаційна модель дозволяє досліджувати вплив помилок на достовірність передавання перестановок. Створені макетні зразки приймально-передавальних пристроїв захищеного інформаційного обміну можуть бути використані в реальних системах бездротового зв'язку. Використання GPU для криптографічних обчислень над перестановками може значно прискорити ефективність обчислення криптографічних операцій з використанням перестановок. Отримані результати є основою для подальшого вдосконалення систем захищеного інформаційного обміну, зокрема розширення трьохетапного криптографічного протоколу на основі перестановок та застосування FPGA з метою підвищення продуктивності роботи під час практичної реалізації.

Основні результати наведених досліджень в цьому розділі представлено в роботах [25], [105], [197].

ВИСНОВКИ

Дисертаційна робота розкриває вирішення актуальної науково-технічної задачі, яка полягає в забезпеченні захищеного інформаційного обміну в системах з нероздільним факторіальним кодуванням зашумленим каналом зв'язку.

До найбільш значимих результатів, отриманих у цій роботі, автором віднесено наступне.

1. Набула подальшого розвитку математична модель процесу виявлення синхрокомбінації систем інформаційного обміну з нероздільним факторіальним кодуванням, що дозволяє оцінити ймовірність встановлення правильної та хибної синхронізації за мажоритарної та кореляційної обробки бітової послідовності синхрокомбінації в ковзному вікні фіксованої довжини та невідомого початкового моменту приймання синхрокомбінації. Застосування розробленої математичної моделі дозволило отримати теоретичну оцінку ймовірності встановлення кадрової синхронізації за невідомого початкового моменту приймання синхрокомбінації. Показники ймовірності кадрової синхронізації демонструють високе значення хибних спрацювань підсистеми синхронізації за фактичної відсутності синхрокомбінації в каналі зв'язку. З метою запобігання хибним спрацюванням і подальшого розвитку методу кадрової синхронізації досліджено величину серії хибних спрацювань підсистеми синхронізації. Під серією хибних спрацювань визначено ситуації, коли підсистема синхронізації фіксує однакове хибне положення межі синхрокомбінації для декількох поспіль зсувів бітів в ковзному вікні, а кількість таких ситуацій приймається, як довжина цієї серії. Величину довжин серій хибних спрацювань визначено шляхом моделювання 10000 випробувань процедури встановлення синхронізації за заданої ймовірності бітової помилки $p_0 = 0.4$ в каналі зв'язку. Для визначеної синхрокомбінації експериментально отримано розподіл довжин серії хибних спрацювань, максимальне значення довжини хибних спрацювань у отриманому розподілі становить $l_{false_synch} = 4$. Отримане

значення використано як поріг серії спрацювань підсистеми синхронізації і зменшує ймовірність спрацювання підсистеми синхронізації за відсутності синхрокомбінації в каналі зв'язку.

2. Розроблено метод кадрової синхронізації на основі ковзного вікна з урахуванням серії спрацювань підсистеми синхронізації. Розроблений метод використовує існуючий метод кадрової синхронізації на основі кореляційної обробки і ковзне вікно фіксованої довжини. З метою запобігання ситуаціям, коли синхрокомбінації не передаються передавачем у канал, використано аналіз довжини серії спрацювань підсистеми синхронізації, що повинно перевищити порогове значення для остаточного встановлення синхронізму. Ефективність роботи розробленого методу перевірено в імітаційній моделі інформаційного обміну шляхом проведення 1000 випробувань з урахуванням аналізу серії спрацювань підсистеми синхронізації та без аналізу серії спрацювань підсистеми синхронізації з метою оцінки впливу точності визначення меж блоків синхронізації і даними. Побудовано імітаційну модель системи інформаційного обміну симплексним двійковим симетричним каналом зв'язку з ймовірністю бітової помилки $p_0 = 0.4$. Наведено структури та алгоритми роботи складових блоків імітаційної моделі: передавача, каналу зв'язку та приймача. Наведено параметри моделювання. Експерименти проведено для визначеного діапазону ймовірності бітової помилки в каналі зв'язку $p_0 = [0.1; 0.4]$ з кроком 0.05. Імітаційне моделювання показало, що аналіз довжини серій спрацювань підсистеми синхронізації зменшує похибку визначення межі між блоком синхронізації L_{block} та блоком даних W_{block} і запобігти хибним спрацюванням підсистеми синхронізації при відсутності синхрокомбінації в каналі зв'язку. Результати роботи моделі демонструють успішне передавання інформації в кожному з 1000 випробувань за ймовірності бітової помилки в каналі зв'язку $p_0 = 0.4$. Запропонований метод кадрової синхронізації на основі ковзного вікна з урахуванням серії спрацювань підсистеми синхронізації може складати основу для реалізації підсистеми

синхронізації системи завадостійкого інформаційного обміну на основі перестановок у реалізації трьохетапного криптографічного протоколу.

3. Розроблено математичну модель скінченного поля квадратних матриць порядку 2 над скінченним полем простих чисел \mathbb{Z}_p . Таке скінченне поле квадратних матриць може бути використане для побудови нових схем криптографічних перетворень. У результаті теоретичного обґрунтування визначено шість сімейств матриць $\Gamma_1 - \Gamma_6$, що є комутативними за операцією множення. Доведено, що розширення множини цих матриць включенням одиничної матриці формує абелеву групу за операцією множення з порядком $p^2 - 1$. Показано, що отримане сімейство матриць $CGL_{b,k}(2, \mathbb{Z}_p)$ є одночасно діагоналізоване. Сформовано поле Галуа порядку p^2 квадратних матриць 2×2 зі звичайними операціями додавання і множення. Продемонстровано, що мультиплікативна група отриманого скінченного поля є циклічною і може використовуватись для організації криптографічного захисту, зокрема для генерації ключових матриць. Розроблено алгоритм перетворення матриці в перестановку шляхом перетворення матриці в десяткову та факторіальну системи числення та програмну модель формування ключів-перестановок через квадратну матрицю. Виконано серію експериментальних досліджень програмної моделі. Визначено що алгоритм перетворення через представлення матриці у десятковій та факторіальній системах забезпечує рівномірний розподіл перестановок на множині можливих значень і може бути використаний для генерації ключових перестановок на основі матричних структур. Результати цього розділу створили теоретичну і практичну основу для інтеграції методів узгодження ключів, представлених у вигляді матриць, із методами факторіального кодування даних у системах захищеного інформаційного обміну, що використовують перестановки як носії інформації.

4. У четвертому розділі дисертаційної роботи представлено комплексне дослідження та практичну реалізацію систем захищеного інформаційного обміну з використанням нероздільного факторіального

кодування даних. Продемонстровано етапи побудови імітаційної моделі системи обміну перестановками в середовищі Simulink. Отримана модель дозволяє оцінити вплив бітової помилки на достовірність передавання перестановок через канал з шумом. Розроблена модель може бути використана для подальшого розвитку та перевірки ефективності методів достовірного передавання, синхронізації на основі нероздільного факторіального кодування даних для реалізації криптографічних протоколів обміну. Створено апаратний макет системи обміну, побудований на базі мікроконтролера nRF52840 та ISM-радіоканалу. Розроблені алгоритми кодування текстових повідомлень у перестановки, які дозволяють стискати двосимвольні текстові повідомлення до 3 байтів замість 8 байтів при довжині перестановки $M = 8$ та представлення символів у форматі ASCII. Макет може використовуватись як прототип для захищених IoT-рішень або навчальний стенд. Досліджено ефективність використання графічних прискорювачів з технологією CUDA для операцій множення над перестановками. Використання алгоритмів множення перестановок на CPU (Ryzen 5 3600) та GPU (Nvidia GTX 1070), показало, що для ефективного прискорення операції множення необхідно щонайменше 6 паралельних операцій. Отримані показники продуктивності операцій множення над перестановками дозволили оцінити за часом можливість проведення атаки грубої сили на трьохетапний протокол з використанням НФКД.

Результати досліджень створюють наукову й практичну основу для розвитку систем захищеного інформаційного обміну НФКД, включаючи вдосконалення алгоритмів синхронізації, інтеграцію апаратних рішень на FPGA та розширення при побудові трьохетапних криптографічних протоколів на основі перестановок. Подальшим перспективним напрямком досліджень є розбудова методів встановлення синхронізації з НФКД, інтеграція матричних криптографічних протоколів з НФКД та розгляд побудови апаратних рішень для практичного застосування у цивільних та військових сферах застосування.

СПИСОК ДЖЕРЕЛ

- [1] A. Jahangeer, S. U. Bazai, S. Aslam, S. Marjan, M. Anas, i S. H. Hashemi, «A Review on the Security of IoT Networks: From Network Layer's Perspective», *IEEE Access*, т. 11, с. 71073-71087, 2023, doi: 10.1109/ACCESS.2023.3246180.
- [2] В. Богом'я, О. Бараненко, і Є. Жуков, «Розвиток постквантової криптографії за рахунок використання алгебри підписів (signature algebra)», *Таврійський науковий вісник*, № 6, с. 11-19, 2024, doi: 10.32782/tnv-tech.2024.6.2.
- [3] Д. Новиков і В. Полторах, «Технології постквантової криптографії», *Адаптивні системи автоматичного управління*, т. 1, № 42, с. 171-183, 2023, doi: 10.20535/1560-8956.42.2023.279169.
- [4] А. Коляда, А. Павлишко, і В. Літвінов, «Криптографія після квантової ери: нові виклики та рішення для інформаційної безпеки», *Informatics & Mathematical Methods in Simulation/Informatika ta Matematičnì Metodi v Modelûvannì*, т. 14, № 3, 2024, doi: 10.15276/imms.v14.no3.183.
- [5] C. Feng і H.-M. Wang, «Secure Short-Packet Communications at the Physical Layer for 5G and Beyond», *IEEE Communications Standards Magazine*, т. 5, № 3, с. 96-102, 2021, doi: 10.1109/MCOMSTD.121.2100028.
- [6] Ch. Rupa, Greeshmanth, і M. A. Shah, «Novel secure data protection scheme using Martino homomorphic encryption», *J Cloud Comp*, т. 12, № 1, с. 47, 2023, doi: 10.1186/s13677-023-00425-7.
- [7] F. Haidary Makoui, T. A. Gulliver, і M. Dakhilalian, «A new code-based digital signature based on the McEliece cryptosystem», *IET Communications*, т. 17, № 10, с. 1199-1207, 2023, doi: 10.1049/cmu2.12607.
- [8] В. Богом'я, Л. Черемісіна, і А. Ярмолатій, «Загрози квантових обчислень для класичних криптографічних алгоритмів», *Загрози квантових обчислень для класичних криптографічних алгоритмів. Водний транспорт*, т. 42, № 1, с. 248-257, 2025, doi: 10.33298/2226-8553.2025.1.42.27.

- [9] H. Huang, C. Li, i L. Deng, «Public-Key Cryptography Based on Tropical Circular Matrices», *Applied Sciences*, т. 12, № 15, 2022, doi: 10.3390/app12157401.
- [10] A. Naseri, A. Abbasi, i R. Atani, «A new public key cryptography using Mq matrix», *Journal of Mathematical Modeling*, т. 11, № 4, с. 681-693, 2023, doi: 10.22124/jmm.2023.23982.2142.
- [11] M. Durcheva i K. Danilchenko, «Secure Key Exchange in Tropical Cryptography: Leveraging Efficiency with Advanced Block Matrix Protocols», *Mathematics*, т. 12, № 10, с. 1429, 2024, doi: 10.3390/math12101429.
- [12] M. Maxrizal, «Public Key Cryptosystem Based on Singular Matrix», *Trends in Sciences*, т. 19, № 3, с. 2147, 2022, doi: 10.48048/tis.2022.2147.
- [13] J. Al-Aazzeh, B. Ayyoub, E. Faure, V. Shvydkiy, O. Kharin, i A. Lavdanskyi, «Telecommunication systems with multiple access based on data factorial coding», *International Journal on Communications Antenna and Propagation*, т. 10, № 2, с. 102-113, 2020, doi: 10.15866/irecap.v10i2.17216.
- [14] E. Faure, A. Shcherba, Y. Vasiliu, i A. Fesenko, «Cryptographic Key Exchange Method for Data Factorial Coding», 2020, с. 643. [Online]. Режим доступу: <https://ceur-ws.org/Vol-2654/paper50.pdf>
- [15] E. V. Faure, «Factorial coding with data recovery», *Bulletin of Cherkasy State Technological University*, № 2, с. 33-39, 2016.
- [16] Б. Ступка, «Методи достовірного передавання інформації в системах з нероздільним факторіальним кодуванням даних за високої ймовірності бітової помилки», PhD Thesis, Черкаський державний технологічний університет, Черкаси, 2024. [Online]. Режим доступу: <https://er.chdtu.edu.ua/handle/ChSTU/4818>
- [17] E. Faure, A. Shcherba, B. Stupka, I. Voronenko, i A. Baikenov, «A Method for Reliable Permutation Transmission in Short-Packet Communication Systems», в *Information Technology for Education, Science, and Technics. ITEST 2022. Lecture Notes on Data Engineering and Communications*

- Technologies*, т. 178, E. Faure, O. Danchenko, M. Bondarenko, Y. Tryus, C. Bazilo, i G. Zaspа, Ред., в *Lecture Notes on Data Engineering and Communications Technologies*, vol. 178. , Cham: Springer Nature Switzerland, 2023, с. 177-195. doi: 10.1007/978-3-031-35467-0_12.
- [18] M. Abu-Faraj, A. Al-Hyari, i Z. Alqadi, «A complex matrix private key to enhance the security level of image cryptography», *Symmetry*, т. 14, № 4, с. 664, 2022, doi: 10.3390/sym14040664.
- [19] F. Al-Shaarani i A. Gutub, «Securing matrix counting-based secret-sharing involving crypto steganography», *Journal of King Saud University - Computer and Information Sciences*. King Saud bin Abdulaziz University, 2021. doi: 10.1016/j.jksuci.2021.09.009.
- [20] T. Kumar i S. Chauhan, «Image cryptography with matrix array symmetric key using chaos based approach», *International Journal of Computer Network and Information Security*, т. 13, № 3, с. 60, 2018, doi: 10.5815/ijcnis.2018.03.07.
- [21] А. Білецький, А. Білецький, i Р. Кандиба, «Матричні аналоги протоколу Діффі-Хеллмана», *Вісник національного університету «Львівська політехніка»*, № 741, с. 128-133, 2012.
- [22] В. Красиленко i Д. Нікітович, «Кооперативний протокол узгодження великорозмірних ізоморфно представлених секретних ключів-перестановок та його моделювання», в *The 9th International scientific and practical conference "Theoretical and practical aspects of the development of science and education"*, Czech Republic., 2024, с. 323. [Online]. Режим доступу: <https://isg-konf.com/theoretical-and-practical-aspects-of-the-development-of-science-and-education/>
- [23] K. Prasad i H. Mahato, «Cryptography using generalized Fibonacci matrices with Affine-Hill cipher», *Journal of Discrete Mathematical Sciences and Cryptography*, т. 25, № 8, с. 2341-2352, 2022, doi: <https://doi.org/10.48550/arXiv.2003.11936>.

- [24] E. V. Faure, V. V. Shvydkyi, A. O. Lavdanskyi, i O. O. Kharin, «Methods of factorial coding of speech signals», *Radio Electronics, Computer Science, Control*, № 4, с. 186-198, 2019, doi: 10.15588/1607-3274-2019-4-18.
- [25] Е. В. Фауре, А. Б. Скуцький, і А. О. Лавданський, «Імітаційна модель системи передавання інформації з нероздільним факторіальним кодуванням даних у середовищі Simulink», *Вісник Черкаського державного технологічного університету*, т. 27, № 4, с. 31-47, 2022, doi: 10.24025/2306-4412.4.2022.273385.
- [26] О. О. Харін, «Методи та засоби інтегрованого захисту інформації в телекомунікаційних системах множинного доступу на основі факторіального кодування даних», PhD Thesis, Черкаський державний технологічний університет, Черкаси, 2020. [Online]. Режим доступу: <https://er.chdtu.edu.ua/handle/ChSTU/1147>
- [27] J. Al-Azzeh, E. Faure, A. Shcherba, i B. Stupka, «Permutation-based frame synchronization method for data transmission systems with short packets», *Egyptian Informatics Journal*, т. 23, № 3, с. 529-545, 2022, doi: 10.1016/j.eij.2022.05.005.
- [28] E. Faure, A. Baikenov, A. Skutskyi, D. Faure, i O. Abramkina, «Algorithms for reliable permutation transmission protocols in noisy communication channels», *CEUR Workshop Proceedings*, т. 3826, с. 40-49, 2024, doi: 10.5281/zenodo.15390412.
- [29] A. Shcherba, E. Faure, i O. Lavdanska, «Three-pass cryptographic protocol based on permutations», в *2020 IEEE 2nd International Conference on Advanced Trends in Information Theory (ATIT)*, 2020, с. 281-284. doi: 10.1109/ATIT50783.2020.9349343.
- [30] P. Du, R. Weber, P. Luszczek, S. Tomov, G. Peterson, i J. Dongarra, «From CUDA to OpenCL: Towards a performance-portable solution for multi-platform GPU programming», *Parallel Computing*, т. 38, № 8, с. 391-407, 2012, doi: <https://doi.org/10.1016/j.parco.2011.10.002>.

- [31] J. Gilger, J. Barnickel, i U. Meyer, «GPU-Acceleration of Block Ciphers in the OpenSSL Cryptographic Library», в *Information Security*, D. Gollmann i F. C. Freiling, Ред., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, с. 338-353. doi: 10.1007/978-3-642-33383-5_21.
- [32] Y. Jiang i M. Lei, «MD5 Calculation and Decryption Using CUDA on GPU», в *Trustworthy Computing and Services: International Conference, ISCTCS 2013, Beijing, China, November 2013, Revised Selected Papers*, Springer, 2014, с. 22-28. doi: 10.1007/978-3-662-43908-1_3.
- [33] J. Gmys, «Optimal Solving of Permutation-based Optimization Problems on Heterogeneous CPU/GPU Clusters», в *Proceedings - 2018 International Conference on High Performance Computing and Simulation, HPCS 2018*, 2018, с. 799-801. doi: 10.1109/HPCS.2018.00129.
- [34] W. Pan, F. Zheng, Y. Zhao, W.-T. Zhu, i J. Jing, «An Efficient Elliptic Curve Cryptography Signature Server With GPU Acceleration», *IEEE Transactions on Information Forensics and Security*, т. 12, № 1, с. 111-122, 2017, doi: 10.1109/TIFS.2016.2603974.
- [35] M. S. Abdulnabi i H. Ahmed, «Design of efficient cyclic redundancy check-32 using FPGA», в *2018 International conference on computer, control, electrical, and electronics engineering (ICCCEEE)*, IEEE, 2018, с. 1-5. doi: 10.1109/ICCCEEE.2018.8515877.
- [36] T. P. Doan i S. Ganesan, «CAN Crypto FPGA Chip to Secure Data Transmitted Through CAN FD Bus Using AES-128 and SHA-1 Algorithms with A Symmetric Key», *SAE Technical Papers*, т. 2017-01-1612, 2017, doi: 10.4271/2017-01-1612.
- [37] G. Leelavathi, K. Shaila, i K. R. Venugopal, «Implementation of Public Key Crypto Processor with Probabilistic Encryption on FPGA for Nodes in Wireless Sensor Networks», в *2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2018*, 2018. doi: 10.1109/ICCCNT.2018.8493894.

- [38] U. Umer, M. Rashid, A. R. Alharbi, A. Alhomoud, H. Kumar, i A. R. Jafri, «An Efficient Crypto Processor Architecture for Side-Channel Resistant Binary Huff Curves on FPGA», *Electronics (Switzerland)*, т. 11, № 7, 2022, doi: 10.3390/electronics11071131.
- [39] L. Gnanasekaran, A. S. Eddin, H. El Naga, i M. El-Hadedy, «Efficient RSA Crypto Processor Using Montgomery Multiplier in FPGA», *Advances in Intelligent Systems and Computing*, т. 1070, с. 379-389, 2020, doi: 10.1007/978-3-030-32523-7_26.
- [40] M. Issad, N. Anane, A. Bellemou, i B. Boudraa, «Secure Hybrid Cryptosystem AES/RSA on FPGA for Data Communication», *Malaysian Journal of Computing and Applied Mathematics*, т. 3, № 1, с. 1-10, 2020, doi: 10.37231/myjcam.2020.3.1.38.
- [41] A. S. Haichour i K. Benfriha, «Empowering Real-Time IoT Applications: A Brief Review on Leveraging GPU Acceleration for Latency Reduction», в *IFIP International Internet of Things Conference*, Springer, 2024, с. 107-120. doi: 10.1007/978-3-031-82065-6_8.
- [42] R. Nair, P. Sharma, i T. Sharma, «Optimizing the performance of IoT using FPGA as compared to GPU», *International Journal of Grid and High Performance Computing (IJGHPC)*, т. 14, № 1, с. 1-15, 2022, doi: 10.4018/IJGHPC.301580.
- [43] Y. Yamato, T. Demizu, H. Noguchi, i M. Kataoka, «Automatic GPU offloading technology for open IoT environment», *IEEE Internet of Things Journal*, т. 6, № 2, с. 2669-2678, 2018, doi: 10.1109/JIOT.2018.2872545.
- [44] W.-K. Lee i S. O. Hwang, «High throughput implementation of post-quantum key encapsulation and decapsulation on GPU for Internet of Things applications», *IEEE Transactions on Services Computing*, т. 15, № 6, с. 3275-3288, 2021, doi: 10.1109/TSC.2021.3103956.
- [45] Y. Huang, Y. Li, Z. Zhang, i R. W. Liu, «GPU-accelerated compression and visualization of large-scale vessel trajectories in maritime IoT industries»,

- IEEE Internet of Things Journal*, т. 7, № 11, с. 10794-10812, 2020, doi: 10.1109/IJOT.2020.2989398.
- [46] R. J. McEliece, «A public-key cryptosystem based on algebraic», *Coding Thv*, т. 4244, № 1978, с. 114-116, 1978.
- [47] F. J. MacWilliams i N. J. A. Sloane, *The theory of error-correcting codes*, т. 16. Elsevier, 1977.
- [48] А. А. Борисенко і А. Е. Горячев, «Исправление ошибок в перестановках», 2013, [Online]. Режим доступа: http://nbuv.gov.ua/UJRN/soi_2013_2_40
- [49] А. А. Борисенко, А. Е. Горячев, Б. К. Лопатченко, і А. Н. Кобяков, «Перестановки в телекоммуникационных сетях», 2013, [Online]. Режим доступа: <http://essuir.sumdu.edu.ua/handle/123456789/33002>
- [50] О. Борисенко і О. Горячев, «Помехоустойчивая передача экономической информации на основе перестановок», *Актуальні проблеми економіки*, № 3, с. 156-163, 2013.
- [51] И. Горбенко, Ю. Стасев, А. Ивашкин, і А. Ткачѳв, «Анализ имитостойкости систем спутниковой связи и управления», *Радиотехника. Харьковский государственный технологический университет радиоэлектроники*, № 112, с. 17-21, 1999.
- [52] И. Горбенко, Ю. Стасев, А. Потий, і А. Ткачев, «Предложения по обеспечению безопасности информации в единой спутниковой системе передачи информации», *Космічна наука і технологія*, т. 6, № 5, с. 62-66, 1998.
- [53] В. Грабчак, «Исследование достоверности передачи данных в АСУВ с использованием каскадных теоретико-кодowych схем», *Системи обробки інформації*, № 9, с. 13-16, 2006.
- [54] В. Грабчак, «Криптоаналіз каскадних теоретико-кодowych схем захисту інформації», 2007.
- [55] В. Грабчак, И. Пасько, Р. Королев, і И. Кужель, «Алгебраическое кодирование алгеброгеометрическими кодами на пространственных кривых», *Системи обробки інформації*, № 8, с. 134-138, 2007.

- [56] А. Кузнецов, «Методика оценки эффективности помехоустойчивого кодирования в каналах с группирующимися ошибками», *Электронное моделирование*, № 3, с. 49-60, 2006.
- [57] А. Кузнецов, «Методика оценки энергетической эффективности двоичных блоковых кодов в каналах с группирующимися ошибками», *Моделювання та інформаційні технології.–К.: НАНУ, ІПМЕ*, № 32, с. 116-124, 2005.
- [58] А. Кузнецов, «Энергетический выигрыш алгеброгеометрического кодирования», 2003.
- [59] А. Кузнецов, Р. Королев, і Ю. Рябуха, «Исследование статистической безопасности генераторов псевдослучайных чисел», *Системи обробки інформації*, № 3, с. 79-82, 2008.
- [60] Е. Онанченко, А. Кузнецов, В. Лысенко, В. Грабчак, і Р. Королёв, «Исследование методов защиты информации, основанных на использовании алгебраических блоковых кодов», *Системи обробки інформації*, № 7, с. 53-58, 2007.
- [61] Е. Л. Онанченко і А. В. Лысенко, «Анализ известных методов декодирования недвоичных блоковых кодов», т. 38, № 3, с. 205-213, 2008.
- [62] А. Стахов, «Компьютеры Фибоначчи и новая теория кодирования: история, теория, перспективы», *Известия Южного федерального университета. Технические науки*, т. 38, № 3, с. 205-213, 2004.
- [63] А. Stakhov, «Fibonacci matrices, a generalization of the “Cassini formula”, and a new coding theory», *Chaos, Solitons & Fractals*, т. 30, № 1, с. 56-66, 2006, doi: 10.1016/j.chaos.2005.12.054.
- [64] А. Stakhov, «The “golden” matrices and a new kind of cryptography», *Chaos, Solitons & Fractals*, т. 32, № 3, с. 1138-1146, 2007, doi: 10.1016/j.chaos.2006.03.069.
- [65] А. Stakhov, V. Massingue, і А. Sluchenkova, «Introduction into Fibonacci coding and cryptography», *Osnova, Kharkov*, т. 5, № 3, 1999.

- [66] В. Чечельницький, «Методологія підвищення ефективності телекомунікаційних систем на основі інтеграції каналного кодування та шифрування даних», дис. д-ра техн. наук, Київ, 2013.
- [67] Н. И. Кушниренко і В. Чечельницький, «Метод криптографической передачи информации на базе эквивалентного класса совершенных двоичных решеток», *Информатика та математичні методи в моделюванні*, № 4, № 3, с. 210-218, 2014.
- [68] М. И. Мазурков, В. Я. Чечельницький, і П. Мурр, «Метод защиты информации на основе совершенных двоичных решеток», *Вісті вищих учбових закладів. Радіoeлектроніка*, т. 51, № 11, с. 53-57, 2008, doi: 10.20535/S0021347008110095.
- [69] М. И. Мазурков, В. Я. Чечельницький, П. Е. Баранов, А. Н. Мелешкевич, С. Н. Кропачев, і Н. И. Кушниренко, «Методы повышения защиты информации путем объединения операций уплотнения, шифрования и канального кодирования», *Вісті вищих учбових закладів. Радіoeлектроніка*, т. 54, № 5, с. 3-16, 2011, doi: 10.20535/S0021347011050013.
- [70] J. Massey, «Optimum frame synchronization», *IEEE transactions on communications*, т. 20, № 2, с. 115-119, 1972.
- [71] M. Chiani і M. G. Martini, «On sequential frame synchronization in AWGN channels», *IEEE Transactions on Communications*, т. 54, № 2, с. 339-348, 2006, doi: 10.1109/TCOMM.2005.863727.
- [72] M. Hasler і T. Schimming, «Chaos communication over noisy channels», *International Journal of Bifurcation and Chaos*, т. 10, № 4, с. 719-735, 2000, doi: 10.1142/S0218127400000505.
- [73] T. Berger, F. Jelinek, і J. Wolf, «Permutation codes for sources», *IEEE Transactions on Information Theory*, т. 18, № 1, с. 160-169, 1972, doi: 10.1109/TIT.1972.1054729.

- [74] D. H. Smith i R. Montemanni, «A new table of permutation codes», *Designs, Codes and Cryptography*, т. 63, № 2, с. 241-253, 2012, doi: 10.1007/s10623-011-9551-8.
- [75] P. J. Cameron, «Permutation codes», *European Journal of Combinatorics*, т. 31, № 2, с. 482-490, 2010, doi: 10.1016/j.ejc.2009.03.044.
- [76] I. F. Blake, G. Cohen, i M. Deza, «Coding with permutations», *Information and Control*, т. 43, № 1, с. 1-19, 1979.
- [77] V. K. Goyal, S. A. Savari, i W. Wang, «On optimal permutation codes», *IEEE Transactions on Information Theory*, т. 47, № 7, с. 2961-2971, 2002, doi: 10.1109/18.959273.
- [78] О. А. Борисенко, О. Є. Горячев, В. В. Сердюк, i М. С. Єрмаков, «Факториальные числа в задачах защиты безопасности», *Безпека інформації*, т. 24, № 3, с. 169-174, 2018, doi: 10.18372/2225-5036.24.13069.
- [79] А. А. Борисенко i А. Е. Горячев, «Обнаружение ошибок на основе перестановок», 2013, [Online]. Режим доступа: https://essuir.sumdu.edu.ua/bitstream-download/123456789/48741/1/Borysenko_Horiachev.pdf
- [80] О. Borysenko *та ін.*, «Factorial numbers and their practical applications», *Applied Sciences*, т. 14, № 19, с. 8588, 2024, doi: 10.3390/app14198588.
- [81] О. Borysenko, О. Horiachev, I. Kulyk, i M. Yakovlev, «Factorial Permutation Generation», в *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, IEEE, 2019, с. 375-378. doi: 10.1109/PICST47496.2019.9061506.
- [82] О. А. Borysenko, О. Y. Horiachev, О. V. Berezhna, S. M. Matsenko, i А. I. Novhorodtsev, «Noise-immune Transfer of Decimal Data with Protection Based on Permutations», в *2023 IEEE 13th International Conference on Electronics and Information Technologies (ELIT)*, Львів: IEEE, 2023, с. 248-251. doi: 10.1109/ELIT61488.2023.10310685.
- [83] А. Borysenko, О. Horiachev, S. Matsenko, i О. Kobiakov, «Noise-immune codes based on permutations», в *2018 IEEE 9th International Conference on*

- Dependable Systems, Services and Technologies (DESSERT)*, Київ: IEEE, 2018, с. 609-612. doi: 10.1109/DESSERT.2018.8409204.
- [84] A. Borysenko і A. Goryachev, «Permutation application in telecommunication systems», в *2012 22nd International Crimean Conference "Microwave & Telecommunication Technology"*, Севастополь: IEEE, 2012, с. 336-337.
- [85] E. Faure, A. Shcherba, і B. Stupka, «Permutation-Based Frame Synchronisation Method for Short Packet Communication Systems», в *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Cracow, Poland: IEEE, 2021, с. 1073-1077. doi: 10.1109/IDAACS53288.2021.9660996.
- [86] E. Faure, V. Shvydkyi, і V. Shcherba, «Combined factorial coding and its properties», *Radio Electronics Computer Science Control*, № 3, с. 80-86, 2016, doi: 10.15588/1607-3274-2016-3-10.
- [87] E. Faure, A. Shcherba, і A. Kharin, «Factorial code with a given number of inversions», *Радіоелектроніка, інформатика, управління*, № 2 (45), с. 143-153, 2018, doi: 10.15588/1607-3274-2018-2-16.
- [88] Е. В. Фауре, «Методологія захисту інформації на основі факторіального кодування даних», дис. д-ра техн. наук, Національний авіаційний університет, Київ, 2018. [Online]. Режим доступу: <http://er.nau.edu.ua/handle/NAU/35990>
- [89] Е. В. Фауре, В. В. Швидкий, і В. О. Щерба, «Метод формирования имитовставки на основе перестановок», *Захист інформації*, т. 16, № 4, с. 334-340, 2014, doi: 10.18372/2410-7840.16.7620.
- [90] Е. В. Фауре, В. В. Швидкий, А. І. Щерба, О. О. Харін, і Б. А. Ступка, «Метод циклової синхронізації на основі перестановок», *Вісник Черкаського державного технологічного університету*, № 4, с. 67-76, 2020, doi: 10.24025/2306-4412.4.2020.222439.
- [91] Е. В. Фауре і Б. А. Ступка, «Імітаційне моделювання процесу встановлення циклового синхронізму в системах зв'язку з нероздільним

- факторіальним кодуванням», *Вісник Черкаського державного технологічного університету*, № 4, с. 16-24, 2021, doi: 10.24025/2306-4412.4.2021.252807.
- [92] Е. В. Фауре і Б. А. Ступка, «Залежність ефективності кадрової синхронізації нероздільних факторіальних кодів від параметрів синхронізації», *Електронне моделювання*, т. 44, № 6, с. 21-35, 2022, doi: 10.15407/emodel.44.06.021.
- [93] Е. В. Фауре, О. О. Харін, і А. О. Лавданський, «Оцінка властивостей синтезованих на основі теорії решіток сигнально-кодових конструкцій для нероздільних факторіальних кодів», *Вісник Черкаського державного технологічного університету*, № 3, с. 40-47, 2020, doi: 10.24025/2306-4412.3.2020.214937.
- [94] Э. В. Фауре, «Факториальное кодирование с восстановлением данных», *Вісник Черкаського державного технологічного університету*, т. 1, № 2, Art. № 2, 2016, doi: 10.24025/2306-4412.2.2016.82932.
- [95] V. Shvydkiy, A. Shcherba, O. Kharin, A. Lavdanskyi, і E. Faure, *Basics theory of inseparable factorial data coding*. Kharkiv: Novyi Kurs, 2021.
- [96] Э. Фауре, В. Швыдкий, і В. Щерба, «Комбинированное факториальное кодирование и его свойства», *Радіоелектроніка, інформатика, управління*, т. 38, № 3, с. 80-86, 2016.
- [97] Е. В. Фауре, В. В. Швидкий, і А. І. Щерба, «Контроль целостности информации на основе факториальной системы счисления», *Journal of Baku engineering university–Mathematics and computer science*, т. 1, № 1, с. 3-13, 2017.
- [98] Е. Фауре, А. Щерба, Б. Ступка, і А. Байкенов, «Метод достовірного передавання перестановок у системах зв'язку з короткими пакетами», на VI Міжнародна науково-практична конференція «Інформаційні технології в освіті, науці і техніці»(ІТОНТ-2022), Черкаси, 2022, с. 220.
- [99] E. Faure, A. Shcherba, B. Stupka, I. Voronenko, і A. Baikenov, «A method for reliable permutation transmission in short-packet communication systems»,

Lecture Notes on Data Engineering and Communications Technologies, с. 177-195, 2022, doi: 10.1007/978-3-031-35467-0_12.

- [100] E. Faure, A. Shcherba, M. Makhynko, C. Bazilo, i I. Voronenko, «Concept for Using Permutation-Based Three-Pass Cryptographic Protocol in Noisy Channels», в *Systems, Decision and Control in Energy V*, A. Zaporozhets, Ред., Cham: Springer Nature Switzerland, 2023, с. 99-113. doi: 10.1007/978-3-031-35088-7_7.
- [101] E. Faure, A. Shcherba, M. Makhynko, B. Stupka, J. Nikodem, i R. Shevchuk, «Permutation-based block code for short packet communication systems», *Sensors*, т. 22, № 14, с. 5391, 2022, doi: 10.3390/s22145391.
- [102] E. Faure, V. Shvidkiy, i A. Shcherba, «Method of forming reproducible and unpredictable sequence of permutations.», *Ukrainian Scientific Journal of Information Security*, т. 20, № 3, 2014, doi: 10.18372/2225-5036.20.7552.
- [103] J. Al-Azzeh, E. Faure, A. Shcherba, i B. Stupka, «Permutation-based frame synchronization method for data transmission systems with short packets», *Egyptian Informatics Journal*, Чер 2022, doi: 10.1016/j.eij.2022.05.005.
- [104] Е. В. Фауре, А. Б. Скуцький, і А. О. Лавданський, «Імітаційна модель передавання текстових і аудіо повідомлень з використанням нероздільного факторіального кодування в середовищі Simulink», в *Challenges and threats to critical infrastructure*, Detroit, Michigan, USA: NGO Institute for Cyberspace Research, 2023, с. 244-246. [Online]. Режим доступу:
<http://repositsc.nuczu.edu.ua/bitstream/123456789/17918/1/Monograph-09-06-2023.pdf>
- [105] А. Лавданський, Е. Фауре, С. Тинимбаєв, і А. Скуцький, «Система захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону», *Вісник Черкаського державного технологічного університету*, т. 27, № 3, с. 41-48, 2022, doi: 10.24025/2306-4412.3.2022.267786.

- [106]О. Харін, «Оцінка властивостей каскадного коду, що поєднує факторіальний та рівноважний код», *Вісник Черкаського державного технологічного університету*, № 2, с. 86-90, 2017.
- [107]А. Нок, «Matrix field theory», 2020. [Online]. Режим доступу: <https://doi.org/10.48550/arXiv.2005.07525>
- [108]D. Serre i D. Serre, *What are matrices*. Springer, New York, 2010.
- [109]D. Bigatti i L. Susskind, «Review of matrix theory», *Strings, Branes and Dualities*, с. 277-318, 1999.
- [110]W. P. Wardlaw, «Matrix representation of finite fields», *Mathematics Magazine*, т. 67, № 4, с. 289-293, 1994, doi: 10.1080/0025570X.1994.11996233.
- [111]L. Brickman, «On the field of values of a matrix», *Proceedings of the American Mathematical Society*, т. 12, № 1, с. 61-66, 1961, doi: 10.2307/2034125.
- [112]R. Bellman, *Introduction to matrix analysis*. SIAM, 1997.
- [113]R. J. McEliece, *Finite fields for computer scientists and engineers*, т. 23. Springer Science & Business Media, 2012.
- [114]M. K. Singh, «Public key cryptography with matrices», в *Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004.*, IEEE, 2004, с. 146-152. doi: 10.1109/IAW.2004.1437810.
- [115]«Звіт про роботу системи виявлення вразливостей і реагування на кіберінциденти та кібератаки», 2022(Q3). [Online]. Режим доступу: <https://scpc.gov.ua/api/docs/4eeb6a10-b7aa-4396-8b04-e0e4b7fca1ba/4eeb6a10-b7aa-4396-8b04-e0e4b7fca1ba.pdf>
- [116]«Деякі питання забезпечення функціонування системи виявлення вразливостей і реагування на кіберінциденти та кібератаки». Постанова Кабінету Міністрів України №1295, 23, Грудень 2020. [Online]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/1295-2020-%D0%BF#Text>
- [117]«Держспецзв'язку: Статистика кібератак за чотири місяці війни», Держспецзв'язку. [Online]. Режим доступу:

<https://www.kmu.gov.ua/news/derzhspecvvyazku-statistika-kiberatak-zachotiri-misyaci-vijni>

- [118] «Кібератаки групи UAC-0118 – дослідження CERT-UA», Державна служба спеціального зв'язку та захисту інформації України. [Online]. Режим доступу: <https://cip.gov.ua/ua/news/kiberataki-grupi-uac-0118-doslidzhennya-cert-ua>
- [119] «Які російські та проросійські хакери атакують Україну», Державна служба спеціального зв'язку та захисту інформації України. [Online]. Режим доступу: <https://cip.gov.ua/ua/news/yaki-rosiiski-ta-prorosiiski-khakeri-atakuyut-ukrayinu>
- [120] Kali Linux. [Online]. Режим доступу: <https://www.kali.org/>
- [121] P. Li *та ін.*, «Scalable parallel ultrafast optical random bit generation based on a single chaotic microcomb», *Light: Science & Applications*, т. 13, № 1, с. 66, 2024, doi: 10.1038/s41377-024-01411-7.
- [122] Ye. Yu. Kaptol і I. D. Horbenko, «Analysis of the possibilities and features of programming cryptology problems on a quantum computer», *Radiotekhnika*, № 202, с. 37-48, 2020, doi: 10.30837/rt.2020.3.202.03.
- [123] S. Joshi, A. K. Bairwa, A. P. Pljonkin, P. Garg, і K. Agrawal, «From Pre-Quantum to Post-Quantum RSA», в *Proceedings of the 6th International Conference on Networking, Intelligent Systems & Security*, 2023, с. 1-8. [Online]. Режим доступу: <https://dl.acm.org/doi/10.1145/3607720.3607721>
- [124] «Are Your Passwords in the Green?», Hivesystems. [Online]. Режим доступу: <https://www.hivesystems.io/blog/are-your-passwords-in-the-green>
- [125] A.-S. Bana, K. F. Trillingsgaard, P. Popovski, і E. de Carvalho, «Short Packet Structure for Ultra-Reliable Machine-Type Communication: Tradeoff between Detection and Decoding», в *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB: IEEE, Квіт 2018, с. 6608-6612. doi: 10.1109/ICASSP.2018.8461650.

- [126]C. Feng, H.-M. Wang, i H. V. Poor, «Reliable and Secure Short-Packet Communications», *IEEE Trans. Wireless Commun.*, т. 21, № 3, с. 1913-1926, Бер 2022, doi: 10.1109/TWC.2021.3108042.
- [127]R. Aleksieieva, A. Fesenko, A. Dudnik, i Y. Zhanerke, «Software Tool for Ensuring Data Integrity and Confidentiality Through the Use of Cryptographic Mechanisms», в *CEUR Workshop Proceedings*, 2023, с. 259-273.
- [128]«Advanced Encryption Standard (AES)». US, 26, Листопад 2001.
- [129]О. Borysenko, О. Berezhna, S. Matsenko, V. Serdiuk, A. Horishniak, i V. Vasilyev, «Нероздільні коди в системах обробки інформації», т. 2, № 64, с. 58-62, 2021.
- [130]О. Харін, «Порівняльна оцінка факторіальних кодів», *Вісник Черкаського державного технологічного університету*, № 4, с. 88-93, 2017.
- [131]Е. Фауре, «Факторіальне кодування з декількома контрольними сумами», *Вісник Житомирського державного технологічного університету*, № 3, с. 104-113, 2016.
- [132]Е. Фауре і О. Харін, «ФАКТОРІАЛЬНЕ КОДУВАННЯ З ВІДНОВЛЕННЯМ ДАНИХ І ВИПРАВЛЕННЯМ ПОМИЛОК», с. 74.
- [133]«Data Encryption Standard (DES)». US, 25, Жовтень 1999.
- [134]W. Diffie i M. Hellman, «New directions in cryptography», *IEEE Trans. Inform. Theory*, т. 22, № 6, с. 644-654, Лис 1976, doi: 10.1109/TIT.1976.1055638.
- [135]«RSA Cryptography Standard». RSA Laboratories, 27, Жовтень 2012.
- [136]T. Elgamal, «A public key cryptosystem and a signature scheme based on discrete logarithms», *IEEE Transactions on Information Theory*, т. 31, № 4, с. 469-472, Лип 1985, doi: 10.1109/TIT.1985.1057074.
- [137]«IEEE Standard Specifications for Public-Key Cryptography», *IEEE Std 1363-2000*, с. 1-228, 2000, doi: 10.1109/IEEESTD.2000.92292.
- [138]N. I. of Standards, Technology (NIST), L. Chen, D. Moody, A. Regenscheid, i A. Robinson, *Digital Signature Standard (DSS)*. Federal Inf. Process. Stds.

- (NIST FIPS), National Institute of Standards and Technology, Gaithersburg, MD, 2023. doi: 10.6028/NIST.FIPS.186-5.
- [139]H. Huang, C. Li, i L. Deng, «Public-Key Cryptography Based on Tropical Circular Matrices», *Applied Sciences*, т. 12, № 15, 2022, doi: 10.3390/app12157401.
- [140]M. Kotov i A. Ushakov, «Analysis of a key exchange protocol based on tropical matrix algebra», *Journal of Mathematical Cryptology*, т. 12, № 3, с. 137-141, 2018, doi: doi:10.1515/jmc-2016-0064.
- [141]D. Rudy i C. Monico, «Remarks on a tropical key exchange system», *Journal of Mathematical Cryptology*, т. 15, № 1, с. 280-283, 2020, doi: 10.1515/jmc-2019-0061.
- [142]B. A. Forouzan, *Cryptography & network security*. McGraw-Hill, Inc., 2007.
- [143]X. Wang i S. Gao, «Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory», *Information Sciences*, т. 507, с. 16-36, 2020, doi: 10.1016/j.ins.2019.08.041.
- [144]F. Dupont, «A new Shamir's three pass random matrix ciphering mechanism», *J Comput Virol Hack Tech*, 2023, doi: 10.1007/s11416-023-00467-0.
- [145]E. Faure, A. Shcherba, M. Makhynko, B. Stupka, J. Nikodem, i R. Shevchuk, «Permutation-Based Block Code for Short Packet Communication Systems», *Sensors*, т. 22, № 14, с. 5391, Лип 2022, doi: 10.3390/s22145391.
- [146]Ю. П. Матурін, Л. І. Комарницька, i І. В. Гордієнко, *Дискретна математика*. ДДПУ ім. І. Франка, 2023.
- [147]E. Faure, A. Shcherba, i B. Stupka, «Permutation-Based Frame Synchronisation Method for Short Packet Communication Systems», в *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Cracow, Poland: IEEE, Бер 2021, с. 1073-1077. doi: 10.1109/IDAACS53288.2021.9660996.

- [148]D. E. Knuth, *The Art of Computer Programming: Introduction to combinatorial algorithms and Boolean functions*, т. 4А. Upper Saddle River, NJ: Addison-Wesley, 2008.
- [149]I. Yoshinori, «Majority circuit», JPH01296825A, 1989
- [150]T. J. Terrell і L.-K. Shark, *Digital Signal Processing*. London: Macmillan Education UK, 1996. doi: 10.1007/978-1-349-13735-0.
- [151]L. Tan і J. Jiang, *Digital Signal Processing. Fundamentals and Applications*, 3rd вид. Elsevier, 2019. doi: 10.1016/C2017-0-02319-4.
- [152]G. Galati, G. Pavan, і C. Wasserzier, «Signal design and processing for noise radar», *EURASIP J. Adv. Signal Process.*, т. 2022, № 1, с. 52, Чер 2022, doi: 10.1186/s13634-022-00884-1.
- [153]H. Chen, G. Aminian, і Y. Bu, «An Algorithm for Computing the Capacity of Symmetrized KL Information for Discrete Channels», 09 2024, с. 01-08. doi: 10.1109/Allerton63246.2024.10735330.
- [154]J. Boiko, I. Pyatin, O. Yeromenko, і D. Shaiuk, «Assessment of the effect of carrier frequency shift on immunity of OFDM telecommunications», *MEASURING AND COMPUTING DEVICES IN TECHNOLOGICAL PROCESSES*, с. 19-26, 2022, doi: 10.31891/2219-9365-2022-71-3-3.
- [155]V. O. Vlasenko, Yu. V. Shchavinskyi, M. M. Zaporozhchenko, і V. S. Tyshchenko, «Analysis of technologies for building a data transmission network with high requirements for information security, reliability and delay», *Zviazok State University of Information and Communication Technologies*, № 3, 2023, doi: 10.31673/2412-9070.2023.032030.
- [156]V. Anand Kumar і V. Nandalal, «A comparative design of 5G communication codes», *International Journal of Communication Systems*, т. 37, № 18. John Wiley and Sons Ltd, 2024. doi: 10.1002/dac.5954.
- [157]D. T. Valentine і B. H. Hahn, *Essential MATLAB for engineers and scientists*, 8ий вид. Academic Press, 2022.
- [158]J. C. da Silva, D. de L. Flor, V. A. de S. Junior, N. S. Bezerra, і A. A. M. de Medeiros, «A Survey of LoRaWAN Simulation Tools in ns-3», *Journal of*

- Communication and Information Systems*, т. 36, № 1, с. 17-30, Лют 2021, doi: 10.14209/jcis.2021.2.
- [159]H. Q. Ta, Q.-V. Pham, K. Ho-Van, i S. W. Kim, «Covert communication with noise and channel uncertainties», *Wireless Networks*, т. 28, № 1, с. 161-172, 2022, doi: 10.1007/s11276-021-02828-3.
- [160]E. Faure, A. Skutskyi, i A. Lavdanskyi, «Algorithms and simulation model for the synchronisation subsystem of the noise-resilient communication system based on permutations», *Вісник Черкаського державного технологічного університету*, т. 4, № 29, с. 62-74, 2024, doi: 10.62660/bcstu/4.2024.62.
- [161]R. L. Rivest, A. Shamir, i L. Adleman, «A method for obtaining digital signatures and public-key cryptosystems», *Commun. ACM*, т. 21, № 2, с. 120-126, 1978, doi: 10.1145/359340.359342.
- [162]A. Shamir, R. L. Rivest, i L. M. Adleman, «Mental Poker», в *The Mathematical Gardner*, D. A. Klarner, Ред., Boston, MA: Springer US, 1981, с. 37-43. doi: 10.1007/978-1-4684-6686-7_5.
- [163]J. K. Massey i J. L. Omura, «Method and apparatus for maintaining the privacy of digital messages conveyed by public transmission», US4567600, 1986
- [164]T. A. Springer, *Linear Algebraic Groups*. Boston, MA: Birkhäuser Boston, 1998. doi: 10.1007/978-0-8176-4840-4.
- [165]A. Baker, *Matrix Groups*. в Springer Undergraduate Mathematics Series. London: Springer, 2002. doi: 10.1007/978-1-4471-0183-3.
- [166]S. Lipschutz, *Schaum's outline of theory and problems of linear algebra*, 2nd ed. в Schaum's outline series. New York: McGraw-Hill, 1991.
- [167]F. R. Gantmacher, *The theory of matrices*, Reprinted., т. 1. Providence, RI: American Mathematical Soc, 1959.
- [168]J. M. Laughlin, «Combinatorial identities deriving from the n-th power of a 2x2 matrix», *Integers*, т. 4, с. 1-15, 2004, doi: 10.48550/ARXIV.1812.11168.
- [169]V. I. Arnold, «Fermat dynamics, matrix arithmetics, finite circles, and finite Lobachevsky planes», *Functional Analysis and Its Applications*, т. 38, № 1, с. 1-13, 2004, doi: 10.1023/B:FAIA.0000024863.06462.68.

- [170]S. Wright, *Quadratic Residues and Non-Residues*, т. 2171. в Lecture Notes in Mathematics, vol. 2171. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-45955-4.
- [171]R. A. Horn і C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 2012.
- [172]K. F. Ireland і M. I. Rosen, *A classical introduction to modern number theory*, 2nd вид. в Graduate texts in mathematics, no. 84. New York: Springer-Verlag, 1990.
- [173]R. Lidl і H. Niederreiter, *Finite fields*, 2ий вид. в Encyclopedia of mathematics and its applications, no. 20. Cambridge: Cambridge University Press, 1997.
- [174]W. H. Greub, *Linear algebra*, т. 23. Springer Science & Business Media, 2012.
- [175]H. Chernoff і E. L. Lehmann, «The Use of Maximum Likelihood Estimates in χ^2 Tests for Goodness of Fit», *The Annals of Mathematical Statistics*, т. 25, № 3, с. 579-586, 1954, doi: 10.1214/aoms/1177728726.
- [176]H. B. Mann і A. Wald, «On the Choice of the Number of Class Intervals in the Application of the Chi Square Test», *The Annals of Mathematical Statistics*, т. 13, № 3, с. 306-317, 1942, doi: 10.1214/aoms/1177731569.
- [177]V. M. Turchyn, «Probability Theory and Mathematical Statistics», 2014, [Online]. Режим доступу: https://mmf.dnu.dp.ua/wp-content/uploads/2020/01/turchynvm_ptams_2014.pdf
- [178]A. Shcherba, E. Faure, A. Skutskyi, і O. Kharin, «Families of Square Commutative 2x2 Matrices», *CEUR Workshop Proceedings*, т. 3550, с. 289-296, 2023, doi: 10.5281/zenodo.15391901.
- [179]E. Faure, A. Shcherba, A. Skutskyi, і A. Lavdanskyi, «A Finite Field of Square Matrices of Order 2», *CEUR Workshop Proceedings*, т. 3550, с. 306-312, 2023, doi: 10.5281/zenodo.15392022.
- [180]E. Faure, A. Shcherba, A. Skutskyi, і A. Lavdanskyi, «A software model to generate permutation keys through a square matrix», *Вісник Черкаського державного технологічного університету*, т. 29, № 2, с. 10-23, 2024, doi: <https://doi.org/10.62660/bcstu/2.2024.10>.

- [181] A. Shcherba, E. Faure, i O. Lavdanska, «Three-Pass Cryptographic Protocol Based on Permutations», в *2020 IEEE 2nd International Conference on Advanced Trends in Information Theory (ATIT)*, 2020, с. 281-284. doi: 10.1109/ATIT50783.2020.9349343.
- [182] «CUDA GPUs | NVIDIA Developer». [Online]. Режим доступу: <https://developer.nvidia.com/cuda-gpus>
- [183] S. A. Manavski, «CUDA Compatible GPU as an Efficient Hardware Accelerator for AES Cryptography», в *2007 IEEE International Conference on Signal Processing and Communications*, 2007, с. 65-68. doi: 10.1109/ICSPC.2007.4728256.
- [184] W. Pan, F. Zheng, Y. Zhao, W.-T. Zhu, i J. Jing, «An Efficient Elliptic Curve Cryptography Signature Server With GPU Acceleration», *IEEE Transactions on Information Forensics and Security*, т. 12, № 1, с. 111-122, 2017, doi: 10.1109/TIFS.2016.2603974.
- [185] J. Gilger, J. Barnickel, i U. Meyer, «GPU-Acceleration of Block Ciphers in the OpenSSL Cryptographic Library», в *Information Security*, D. Gollmann i F. C. Freiling, Ред., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, с. 338-353. doi: 10.1007/978-3-642-33383-5_21.
- [186] «google/benchmark: A microbenchmark support library». [Online]. Режим доступу: <https://github.com/google/benchmark>
- [187] A. O. Lavdanskyi, E. V. Faure, i V. O. Shcherba, «Increasing the speed of the permutations multiplication operation due to use of SIMD instructions», *Visnyk Cherkaskogo derzhavnogo tehnologichnogo universitetu*, № 3, с. 36-43, 2021, doi: 10.24025/2306-4412.3.2021.245347.
- [188] J. S. Al-Azzeh, B. Ayyoub, E. Faure, V. Shvydkyi, O. Kharin, i A. Lavdanskyi, «Telecommunication Systems with Multiple Access Based on Data Factorial Coding», *International Journal on Communications Antenna and Propagation (IRECAP)*, т. 10, № 2, с. 102, Kbit 2020, doi: 10.15866/irecap.v10i2.17216.

- [189]J. Al-Azzeh, E. Faure, A. Shcherba, i B. Stupka, «Permutation-based frame synchronization method for data transmission systems with short packets», *Egyptian Informatics Journal*, Чеп 2022, doi: 10.1016/j.eij.2022.05.005.
- [190]E. Faure, A. Shcherba, Y. Vasiliu, i A. Fesenko, «Cryptographic Key Exchange Method for Data Factorial Coding», т. 2654, с. 643, 08 2020.
- [191]E. Faure, A. Shcherba, i B. Stupka, «Permutation-Based Frame Synchronisation Method for Short Packet Communication Systems», в *2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Cracow, Poland: IEEE, Бep 2021, с. 1073-1077. doi: 10.1109/IDAACS53288.2021.9660996.
- [192]U. Umer, M. Rashid, A. R. Alharbi, A. Alhomoud, H. Kumar, i A. R. Jafri, «An Efficient Crypto Processor Architecture for Side-Channel Resistant Binary Huff Curves on FPGA», *Electronics (Switzerland)*, т. 11, № 7, 2022, doi: 10.3390/electronics11071131.
- [193]G. Leelavathi, K. Shaila, i K. R. Venugopal, «Implementation of Public Key Crypto Processor with Probabilistic Encryption on FPGA for Nodes in Wireless Sensor Networks», в *2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2018*, 2018. doi: 10.1109/ICCCNT.2018.8493894.
- [194]M. Kashif i İ. Çiçek, «Field-programmable gate array (FPGA) hardware design and implementation of a new area efficient elliptic curve crypto-processor», *Turkish Journal of Electrical Engineering and Computer Sciences*, т. 29, № 4, с. 2127, 2021, doi: 10.3906/ELK-2008-8.
- [195]L. Gnanasekaran, A. S. Eddin, H. El Naga, i M. El-Hadedy, «Efficient RSA Crypto Processor Using Montgomery Multiplier in FPGA», *Advances in Intelligent Systems and Computing*, т. 1070, с. 379-389, 2020, doi: 10.1007/978-3-030-32523-7_26.
- [196]T. P. Doan i S. Ganesan, «CAN Crypto FPGA Chip to Secure Data Transmitted Through CAN FD Bus Using AES-128 and SHA-1 Algorithms with A

Symmetric Key», *SAE Technical Papers*, т. 2017-March, № March, 2017, doi: 10.4271/2017-01-1612.

- [197] A. Lavdanskyi, E. Faure, A. Skutskyi, i C. Bazilo, «Accelerating Operations on Permutations Using Graphics Processing Units», *Lecture Notes on Data Engineering and Communications Technologie*, т. 178, с. 3-12, 2023, doi: 10.1007/978-3-031-35467-0_1.

ДОДАТКИ

Додаток А. Лістинги розрахунково-експериментальних моделей

А.1. Лістинг імітаційно-програмної моделі інформаційного обміну

з НФКД на мові Matlab

```

clear;
sL = [0 1 7 3 2 5 4 6];
M = 8;
lr = 3;
permLen = M*lr;
% noiseProbability = 0.4; % commented for experiment stats
numL = 75;
K = 1; % not implemented
numW = 89;
rxSynchTreshold = -1;

sLlenght = permLen * numL;
sWlenght = permLen * (M * lr - 1);
windowSlSize = permLen * numL * K;
dLim = fix((((M * lr) / 2) - 1) / 2);

binSl = binArray(dec2bin(sL, lr)); % binary presentation of SL
sLShifts = genShifts(binSl, permLen-1); % generate bit shifts of SL and save in 2D
array
lettersFrameLenght = permLen * numL * K;
wordsFrameLenght = sWlenght * numW;

% letters+words;
frameLenght = lettersFrameLenght + wordsFrameLenght;

% concat letters in serial SL0-SL0-SL0-...
synchLetters = boolean(zeros(1, lettersFrameLenght));
for idx = 1:numL * K
    start = 1 + (idx - 1) * permLen;
    stop = (start - 1) + permLen;
    synchLetters(start:stop) = binSl;
end

% concat one word in serial SL1-SL2-...SL23
word = boolean(zeros(1, sWlenght));
for shift = 1:(permLen - 1)
    start = 1 + (shift - 1) * permLen;
    stop = start + permLen - 1 ;
    word(start:stop) = sLShifts(shift + 1, :);

```

end

```
% concat words in serial SL1-...-SL23-SL1-...-SL23
synchWords = boolean(zeros(1, wordsFrameLenght));
```

```
for idx = 1:numW
    start = 1 + ((idx - 1) * sWlenght);
    stop = start + sWlenght - 1;
    synchWords(start:stop) = word;
end
```

end

```
% rx model
```

```
isDataRecieved = false;
bitShifts = 0;
rxSynchNums = 0;
currentWord = [];
wordPhaseShift = 0;
```

```
% tranciever output sLlenght bits of 0 + sLlenght of synchLetters + synchWords
txOutput = [boolean(zeros(1, sLlenght)) synchLetters synchWords];
```

```
% clear experiment stats
```

```
experimentResult = [];
totalResults = [];
totalFalseBit = [];
totalTimes = [];
```

```
or noiseProbability=0.4:-0.05:0.1
```

```
falseBitinWord = "null";% experiment stats
experimentResult = noiseProbability;% experiment stats
experimentFalseBit = noiseProbability;% experiment stats
expTime = noiseProbability;% experiment stats
```

```
for experiment=1:1000
```

```
tic % experiment stats
```

```
while(~isDataRecieved)
```

```
    if(bitShifts < (frameLenght))
```

```
        % apply noise by bsc - binary symetric channel
```

```
        % window of rx is static, so we just shifting data to the left <--
```

```
        channelData = circshift(bsc(txOutput, noiseProbability), -bitShifts, 2); % 2 - it's
```

```
shift row
```

```
    else
```

```
        disp("End of packet reached!")
```

```
        disp("RX side didn't receive data.")
```

```
        break;
```

```
    end
```

```
    bitShifts = bitShifts + 1;
```

```

% reciever logic
isWord = false;
prefix = majProc(channelData(1:lettersFrameLenght), permLen);%take frame
size from channel
[isSynch, ~, shift] = checkShift(prefix, sLShifts, dLim);

% check synch
if(isSynch)

    % check treshhold value of sych
    if(rxSynchNums > rxSynchTreshold)
        phaseShift = permLen - shift; % number of skipped bits
        if(shift > 0)
            bitShifts = bitShifts + phaseShift;
            channelData = circshift(channelData, -phaseShift, 2);
        end

        % get word (1:M*lr-1 shifts )
        currentWord = checkWords(channelData, numL, numW, M, lr, sLShifts,
dLim);
        % we get quantity of letters in word
        if(size(currentWord,2) == M*lr-1)
            % check if the all letters in word are different
            if(isPerm(currentWord))
                % check if the first letter in word is 1
                if(currentWord(1) == 1)
                    % end of RX cycle
                    isDataRecieved = true;
                    disp("RX side recieved data.")
                    % statistics
                    falseBitinWord = sLlenght - (bitShifts - 1);
                else
                    % statistics
                    wordPhaseShift = wordPhaseShift + 1;
                end
            else
                disp("RX'ed data is incorrect.")
                currentWord = [];
            end
        end
    end

else
    % inc value of sych

```

```

        rxSynchNums = rxSynchNums + 1;
    end
else
    % reset treshold value of sych
    rxSynchNums = 0;
end
end
t = toc;% experiment stats
experimentFalseBit = [experimentFalseBit falseBitinWord];% experiment stats
experimentResult = [experimentResult isDataRecieved];% experiment stats
expTime = [expTime t];% experiment stats
isDataRecieved = false;% experiment stats
bitShifts = 0;% experiment stats
rxSynchNums = 0;% experiment stats
currentWord = [];% experiment stats
end
totalResults = [ totalResults experimentResult'];% experiment stats
totalFalseBit = [totalFalseBit experimentFalseBit'];% experiment stats
totalTimes = [totalTimes expTime'];% experiment stats
end

% check words
function currentWord = checkWords(fragment, letters, words, M, lr,
correctionTable, dLim)
% take all letters wrom word, do maj. processing and fixing corrupted data
% return list with idx of letters. [1,2,..n]
currentWord = [];
permLen = M*lr;
lettersFrameLenght = letters * permLen;
sWlenght = permLen * (permLen - 1);
for k=1:(permLen-1)
    %take all sW from numSW
    wordParticleBuffer = [];
    % collect all parts of the words
    for i=1:words
        start = lettersFrameLenght + permLen*(k-1) + sWlenght*(i-1) + 1;
        stop = lettersFrameLenght + permLen*(k-1) + sWlenght*(i-1) + permLen;
        wordParticleBuffer = [wordParticleBuffer fragment(start:stop)];
    end
    % correct data if dLim true
    [isData, ~,dataCorrectionIdx] = checkShift(majProc(wordParticleBuffer,
permLen), correctionTable((2:end),:), dLim) ;
    if(~isData)
        isWord = false;
    end
end

```

```

        currentWord = [];
        return
    else
        isWord = true;
        data = correctionTable(dataCorrectionIdx + 2,:);
    end
    if(isWord)
        % save letter of word
        currentWord = [currentWord dataCorrectionIdx + 1];
    end
end
end
end

```

```

function CkSt = isPerm (data_in)
M=length(data_in);
temp = zeros(M,1);
counter = 0;
for i=0:M
    temp(:,1) = data_in==i; % порівнюємо вхідне повідомлення з М можливими
    if(sum(temp,"all","default")>1)% якщо поточний і зустрівся більше 1 разу
        counter=counter+1;% фіксуємо це
    end
end
if counter==0% якщо повторень немає
    CkSt = true;% це перестановка
else
    CkSt = false;% є повторення - не перестановка
end
end
end

```

```

function permutation = bin2perm(binaryArray,bitsPerSymbol)
%BIN2PERM convert binary array to permutation
binaryLenght = length(binaryArray);
permutationLenght = binaryLenght/bitsPerSymbol;
permutation = zeros(1, permutationLenght);
for idx=1:permutationLenght-1
    start = 1+(idx-1)*bitsPerSymbol;
    stop = start + bitsPerSymbol - 1;
    permutation(idx) = bin2dec(int2str(binaryArray(start:stop)));
end
return
end

```

```

function shifts = genShifts(binaryData, nShifts)
shifts = zeros(nShifts, size(binaryData,2));

```

```
% first element is the original
shifts(1, :) = binaryData;
for shift = 1:nShifts
    shifts(shift+1, :) = circshift(binaryData,-shift, 2);
end
shifts = boolean(shifts);
return
end
```

A.2. Лістинг імітаційно-програмної моделі визначення величини серії хибних спрацювань підсистеми синхронізації з НФКД на мові Matlab

```

clear;
sL = [0 1 7 3 2 5 4 6];
M = 8;
lr = 3;
permLen = M*lr;

numL = 75;
K = 1;
numW = 89;

sLlenght = permLen * numL;
sWlenght = permLen * (M * lr - 1);
windowSlSize = permLen * numL * K;
dLim = fix((((M*lr)/2)-1)/2);
binSl = binArray(dec2bin(sL, lr)); % binary presentation of SL
sLShifts = genShifts(binSl, permLen-1);
lettersFrameLenght = permLen * numL * K;
wordsFrameLenght = sWlenght * numW;

% letters+words; 24 * 75 * 1 + 24 * 24-1 * 89 = 50928
frameLenght = lettersFrameLenght + wordsFrameLenght;

% concat letters in serial
synchLetters = boolean(zeros(1, lettersFrameLenght));
for idx = 1:numL*K
    start = 1 + (idx - 1) * permLen;
    stop = (start - 1) + permLen;
    synchLetters(start:stop) = binSl;
end

% concat one word in serial
word = boolean(zeros(1, sWlenght));
for shift = 1:(permLen-1)
    start = 1 + (shift-1)*permLen;
    stop = start + permLen - 1 ;
    word(start:stop) = sLShifts(shift+1, :);
end

% concat words in serial

```



```

else
falseSynchNums = 1;
end

detectedShifts = detectedShifts + ", " + mod(shift-1, permLen) + " ";
end
else
if(falseSynchNums > 0)
totalFalseSynch = [totalFalseSynch falseSynchNums] ;
totalDetectedShifts = [totalDetectedShifts detectedShifts];

falseSynchNums = 0;
detectedShifts = "";
end
if(trueSynchNums > 0)
totalTrueSynch = [totalTrueSynch trueSynchNums] ;
trueSynchNums = 0;
end
% display("No synch on " + (shift-1) + " shift.")
end

end
if(exp == 1)
experimentsDetectedShifts(exp) = size(totalDetectedShifts, 2)
else
experimentsDetectedShifts(exp) = size(totalDetectedShifts, 2) -
experimentsDetectedShifts(exp-1);
end

end

function shifts = genShifts(binaryData, nShifts)
shifts = zeros(nShifts, size(binaryData,2));
% first element is the original
shifts(1, :) = binaryData;
for shift = 1:nShifts
shifts(shift+1, :) = circshift(binaryData,-shift, 2);
end
shifts = boolean(shifts);
return
end

% функція обробки блоків за мажоритарним принципом
% вхід - фрагмент з l блоків бітів (data); довжина фрагменту l (blockLen)
% вихід - l фрагмент уточненої послідовності

```

```

function R = majProc(data, blockLen)
R = boolean(zeros(1, blockLen));
numOnes = 0;
numZeros = 0;
lFragments = fix(size(data,2)/blockLen);
for bit = 1:blockLen
    for lIdx = 0:lFragments-1
        if(data(lIdx*blockLen+bit))
            numOnes = numOnes + 1;
        else
            numZeros = numZeros + 1;
        end
    end
    if(numOnes > numZeros)
        R(bit) = true;
    end
    numOnes = 0;
    numZeros = 0;
end
return
end

```

```

function result = perm2binary(p, l)
    % Ініціалізуємо порожній логічний вектор
    result = false(length(p) * l, 1);

    % Проходимо по кожному елементу вхідного вектора p
    for i = 1:length(p)
        % Перетворюємо число в двійковий рядок заданої довжини
        binaryStr = dec2bin(p(i), l);

        % Перетворюємо двійковий рядок в логічний масив
        binaryArray = binaryStr == '1';

        % Вставляємо логічний масив у відповідну позицію в результаті
        result((i-1)*l+1:i*l) = binaryArray;
    end
    result = result';
    return
end

```

```

function [outputArray] = binArray(data)
len = size(data, 2);
terms = size(data, 1);

```

```
outputArray =boolean(zeros(1, len*terms));  
for term=1:terms  
    for bit = 1:len  
        outputArray(bit+((term-1)*len)) = data(term, bit) == '1';  
    end  
end  
return  
end
```

А.3. Лістинг імітаційно-програмної моделі перетворення квадратної матриці другого порядку в перестановку на мові Matlab

```

pMax = 17; %maximum p
pMin = 0; %minimum p
M = 8; % permutation lenght
Mmax = factorial(M); %максимальна кількість чисел при заданій довжині
Amax = fac2int(reshape((M-1:-1:0),[M 1])); %максимальне число при заданій
довжині

nRows = 2; %rows
nCols = 2; %columns

nCollect = 1;%індекс вибірки
end_reached = 0;%прапор проходу всіх значень вибірки

%формування матриці перебором всіх допустимих значень
A = zeros(nRows,nCols);
%формування додаткової матриці для варіанту b)
matrixCoefficients = (0:M-1);
sizeOfCoefficients = size(matrixCoefficients,2);
%виділення пам'яті для майбутніх коефіцієнтів на основі вихідних даних
newAmatrix = zeros(nRows,nCols,(ceil(sizeOfCoefficients/(nRows*nCols))));
%заповнюємо нові матриці по порядку (зліва на право)
% у випадку не співпадіння кількості нових комірок і макс кількістю коеф.
% коефіцієнти замикаються в кільце 0-M-0
quantityNewAmatrix = size(newAmatrix,3);
cursor = 1;% позиція вибірки коефіцієнтів з matrixCoefficients
for currMartix = 1:quantityNewAmatrix
    for currRow = 1:nRows
        for currCols = 1:nCols
            newAmatrix(currRow,currCols,currMartix) = matrixCoefficients(cursor);
            if cursor < M
                cursor = cursor + 1;
            else
                cursor = 1;
            end
        end
    end
end
disp(newAmatrix)
while end_reached == 0% цикл для статистики
    aCollection(:,nCollect) = A;%збір для статистики

```

```

Adec = 0;
rankP = 0;
for row = 1:nRows
    for col = 1:nCols
        %      clc
        %      disp("Коефіцієнт " + A(row,col) + " взятий в степені "...
        %          +rankP+ " значення P = " +pMax)
        Adec = Adec + A(row,col)*(pMax^rankP);
        rankP = rankP + 1;
    end
end
AdecCollection(:,nCollect) = Adec;%збір для статистики
%disp("Отримане число з матриці A = " + Adec)

AmodM = mod(Adec,Mmax);
%disp("Число отримане з матриці за модулем M! = " + AmodM)
AmodMCollection(:,nCollect) = AmodM; %збір для статистики
% ApermMinBase = int2fac(AmodM,0);
% permutMinBase = size(ApermMinBase,1);
% ApermMin = reshape(ApermMinBase,[1 permutMinBase]);
% disp("Число (Adec)mod(M!) = " + AmodM + " з мінімальною кількістю
розрядів перестановки "...
%   + permutMinBase + " дорівнює:")
% disp(ApermMin)

ApermMinBase = int2fac(AmodM,M);
permutMinBase = M;
ApermMin = reshape(ApermMinBase,[1 permutMinBase]);
% disp("Число (Adec)mod(M!) = " + AmodM + " з заданою кількістю розрядів
перестановки "...
%   + permutMinBase + " дорівнює:")
% disp(ApermMin)
ApermMinCollection(:,nCollect) = ApermMin;%збір для статистики
%fac2int(ApermMinBase)
%-----b варіант -----

%множення матриці A на матрицю нових коефіцієнтів
b = pagemtimes(A, newAmatrix);
%b = A .* newAmatrix;
% c(:,1) = A * newAmatrix(:,1);
% c(:,2) = A * newAmatrix(:,2);
% b = c;
b = reshape(b,[1 quantityNewAmatrix*nRows*nCols]);
%
temp_item = 0;

```

```

for rank = 1:M
    temp_item = b(rank);
    b(rank) = mod(temp_item,rank);%тут на виході не перестановка, а лише
    коефіцієнти..
end
BpermutM = wrev(b);
Bdec = fac2int(reshape(BpermutM,[M 1]));%-----<LOOKUP!

BdecCollection(:,nCollect) = Bdec;%збір для статистики за варіантом
перетворення б)
BpermutMCollection(:,nCollect) = BpermutM;%факторіальне представлення
числа
nCollect = nCollect + 1;

currCol = 1;
currRow = 1;
if A(currRow,currCol) < pMax-1
    A(currRow,currCol) = A(currRow,currCol) + 1;
else
    A(currRow,currCol) = 0;
    if A(currRow,currCol+1) < pMax-1
        A(currRow,currCol+1) = A(currRow,currCol+1)+1;
    else
        A(currRow,currCol+1) = 0;
        if A(currRow+1,currCol) < pMax-1
            A(currRow+1,currCol) = A(currRow+1,currCol)+1;
        else
            A(currRow+1,currCol) = 0;
            if A(currRow+1,currCol+1) < pMax-1
                A(currRow+1,currCol+1) = A(currRow+1,currCol+1)+1;
            else
                end_reached = 1;%точка виходу, всі варіанти опрацьовані
                break;
            end
        end
    end
end
end
end
end

%Статистика:
%-----
binrangesA = (0:1:max(AmodMCollection));
bincountsA = histc(AmodMCollection, binrangesA);
%-----
binrangesB = (0:1:max(BdecCollection));
bincountsB = histc(BdecCollection, binrangesB);

```

```

statA = datastats(AmodMCollection')
despersiaA = var(AmodMCollection)
statB = datastats(BdecCollection')
despersiaB = var(BdecCollection)
figure('Name','Absolute frequency (first algorithm)','NumberTitle','off');
barAvar = bar(binrangesA,bincountsA,'hist')
xlabel('Numbers after matrix transformation');
ylabel('Absolute frequency')
xlim([0 max(binrangesA)*1.05])
ylim([0 max(bincountsA)*1.05])

figure('Name','Absolute frequency (second algorithm)','NumberTitle','off');
barBvar = bar(binrangesB,bincountsB,'hist')
xlabel('Numbers after matrix transformation');
ylabel('Absolute frequency')
xlim([0 max(binrangesB)*1.05])
ylim([0 max(bincountsB)*1.05])

fiCumulativeA = bincountsA/sum(bincountsA);
fiAplot = figure('Name','Relative frequency (first algorithm)','NumberTitle','off');
hold on
plot(binrangesA,fiCumulativeA,'LineStyle','-','Color','b');
stem(statA.median,max(fiCumulativeA),"filled",'LineStyle',':', ...
      'Marker','*','Color','r');
text((statA.median),max(fiCumulativeA),"Median","Color", ...
      "r","VerticalAlignment","bottom");
hold off
xlabel('Numbers after matrix transformation');
ylabel('Relative frequency (first algorithm)')
xlim([0 (max(AmodMCollection))]);
ylim([0 (max(fiCumulativeA)*1.05)]);

fiCumulativeB = bincountsB/sum(bincountsB);
fiBplot = figure('Name','Relative frequency (second
algorithm)','NumberTitle','off');
hold on
plot(binrangesB,fiCumulativeB,'LineStyle','-','Color','b');
stem(statB.median,max(fiCumulativeB),"filled",'LineStyle',':', ...
      'Marker','*','Color','r');
text((statB.median),max(fiCumulativeB),"Median","Color", ...
      "r","VerticalAlignment","bottom");
hold off
xlabel('Numbers after matrix transformation');
ylabel('Relative frequency (second algorithm)')

```



```

xlim([0 max(BdecCollection)]);
ylim([0 (max(fiCumulativeB)*1.05)]);

maxRepA = max(bincountsA)
minRepA = min(bincountsA)
maxRepB = max(bincountsB)
[elem,ind]=find(bincountsB~=0);
minRepB = min(bincountsB(ind))
wiAmax = max(fiCumulativeA)
wiAmin = min(fiCumulativeA)
[elem, pos]=find(fiCumulativeB~=0)
wiBmax = max(fiCumulativeB)
wiBmin = min(fiCumulativeB(pos))
[el,pos]=find(bincountsB==minRepB)
function num = fac2int (factArray)%колонки - перестановки, рядки - розряди
rankArray = size(factArray, 1)-1;% визначаємо розрядність у факторіальному
представленні
if rankArray == 0
    disp("Помилка перетворення, вхідний формат не відповідає вимогам -
колонки" + ...
        " це перестановки, рядки - розряди!")
    num = -1;
    return
end
permutPoll = size(factArray, 2);
tempNum = zeros(permutPoll,1);%тимчасовий буфер результату
for currentPerm = 1 : permutPoll
    i = 0;
    tRankArray = rankArray;
    while(tRankArray>=0)
        tempNum(currentPerm,1) = tempNum(currentPerm,1) +
factArray(tRankArray+1, currentPerm)*factorial(i);
        i = i+1;
        tRankArray = tRankArray-1;
    end
end

end
num = tempNum;%вихід рядки це перетворені числа
end

function factorialValue = int2fac(intValue, m)
%перевірка чи задана розрядність задовольняє вхідне значення
if(m)
    base = m;
    probablyMax = 0;

```

```

while m>0
    m = m-1;
    probablyMax = probablyMax + m * factorial(m);
end
if(probablyMax<intValue)
    disp("Помилка! Кількість розрядів недостатньо для представлення
необхідного числа у ФК")
    return
end

else
    base = m;
    probablyMax = 0;
    while 1
        probablyMax = probablyMax + base * factorial(base);
        if(probablyMax < intValue)
            base = base + 1;
        else
            base = base + 1;
            break
        end
    end
end
nextIntValue = 0;
currentIntValue = intValue;
factorialValue=zeros(base, 1);
for basedRank = 1:base
    nextIntValue = fix(currentIntValue/basedRank);%ціла частина
    factorialValue(base - (basedRank-1) ,:)=mod(currentIntValue , basedRank);%
лишок від ділення
    currentIntValue = nextIntValue;
end
end

```

**Додаток Б. Список публікацій здобувача за темою дисертації та відомості
про апробацію результатів дисертації**

Наукові праці, в яких опубліковані основні наукові результати дисертації

- [1] E. Faure, A. Baikenov, A. Skutskyi, D. Faure, i O. Abramkina, «Algorithms for reliable permutation transmission protocols in noisy communication channels», *CEUR Workshop Proceedings*, т. 3826, с. 40-49, 2024, doi: [10.5281/zenodo.15390412](https://zenodo.org/record/15390412) (Scopus)
- [2] E. Faure, A. Shcherba, A. Skutskyi, i A. Lavdanskyi, «A Finite Field of Square Matrices of Order 2», *CEUR Workshop Proceedings*, т. 3550, с. 306-312, 2023, doi: [10.5281/zenodo.15392022](https://zenodo.org/record/15392022) (Scopus)
- [3] E. Faure, A. Shcherba, A. Skutskyi, i A. Lavdanskyi, «A software model to generate permutation keys through a square matrix», *Вісник Черкаського державного технологічного університету*, т. 29, № 2, с. 10-23, 2024, doi: [10.62660/bcstu/2.2024.10](https://bcstu/2.2024.10)
- [4] E. Faure, A. Skutskyi, i A. Lavdanskyi, «Algorithms and simulation model for the synchronisation subsystem of the noise-resilient communication system based on permutations», *Вісник Черкаського державного технологічного університету*, т. 4, № 29, с. 62-74, 2024, doi: [10.62660/bcstu/4.2024.62](https://bcstu/4.2024.62)
- [5] A. Shcherba, E. Faure, A. Skutskyi, i O. Kharin, «Families of Square Commutative 2x2 Matrices», *CEUR Workshop Proceedings*, т. 3550, с. 289-296, 2023, doi: [10.5281/zenodo.15391901](https://zenodo.org/record/15391901) (Scopus)
- [6] A. Lavdanskyi, E. Faure, A. Skutskyi, i C. Bazilo, «Accelerating Operations on Permutations Using Graphics Processing Units», *Lecture Notes on Data Engineering and Communications Technologie*, т. 178, с. 3-12, 2023, doi: [10.1007/978-3-031-35467-0_1](https://doi.org/10.1007/978-3-031-35467-0_1) (Scopus)
- [7] Е. В. Фауре, А. Б. Скуцький, і А. О. Лавданський, «Імітаційна модель системи передавання інформації з нероздільним факторіальним кодуванням даних у середовищі Simulink», *Вісник Черкаського*

державного технологічного університету, т. 27, № 4, с. 31-47, 2022, doi: [10.24025/2306-4412.4.2022.273385](https://doi.org/10.24025/2306-4412.4.2022.273385)

- [8] А. О. Лавданський, Е.В. Фауре, С. Т. Тинимбаєв, і А. Б. Скуцький, «Система захищеного інформаційного обміну текстовими даними через радіоканал ISM-діапазону», *Вісник Черкаського державного технологічного університету*, т. 27, № 3, с. 41-48, 2022. doi: [10.24025/2306-4412.3.2022.267786](https://doi.org/10.24025/2306-4412.3.2022.267786)

Наукові праці, які засвідчують апробацію матеріалів дисертації

- [1] Е. В. Фауре і А. Б. Скуцький, «Розробка моделі трьохетапного криптографічного протоколу на основі перестановок», в *Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей XII Міжнародної науково-технічної конференції, Баку–Харків–Жиліна, 27–28 квітня 2022 року*, Харків: ХНУРЕ, 2022, с. 138. [Online]. Режим доступу: https://nure.ua/wp-content/uploads/conf-2022-akov/telecom_2022_volume_1.pdf
- [2] Е. В. Фауре, А. Б. Скуцький, А. О. Лавданський, і О. О. Харін, «Протокол надійного передавання перестановок в умовах інтенсивних шумів у каналі зв'язку», в *Інновації та перспективні шляхи розвитку інформаційних технологій (ІПШРІТ-2024): тези доповідей III Міжнародної науково-практичної інтернет-конференції*, Черкаси: ЧДТУ, 2024, с. 107. [Online]. Режим доступу : https://drive.google.com/file/d/15-8DffQpER_5F6TniHYNIDf2BjOPjehX/view?usp=drive_link

Наукові праці, які додатково відображають наукові результати дисертації

- [1] Е. В. Фауре, А. Б. Скуцький, і А. О. Лавданський, «Імітаційна модель передавання текстових і аудіо повідомлень з використанням нероздільного факторіального кодування в середовищі Simulink», в

Challenges and threats to critical infrastructure, Detroit, Michigan, USA: NGO Institute for Cyberspace Research, 2023, с. 244-246. [Online]. Режим доступу: <https://er.chdtu.edu.ua/bitstream/ChSTU/4539/1/Monograph-09-06-2023-Faure2.pdf>

Апробацію результатів дисертації проведено на:

- XII Міжнародній науково-технічній конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» (Баку–Харків–Жиліна, 27–28 квітня 2022 року);
- Міжнародна науково-практична конференція «Information Technology for Education, Science and Technics» (ITEST 2022);
- II Міжнародна науково-практична конференція «Виклики та загрози об'єктам критичної інфраструктури» (Київ, 29-30 червня, 2023);
- Cybersecurity Providing in Information and Telecommunication Systems II (CPITS-II-2023), (Kyiv, 26 October 2023);
- Cybersecurity Providing in Information and Telecommunication Systems II (CPITS-II 2024), (Kyiv, 26 October 2024);
- III Міжнародній науково-практичній інтернет-конференції «Інновації та перспективні шляхи розвитку інформаційних технологій» (ІПШРІТ-2024), (Черкаси, 22 листопада 2024 року).