Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації

КОМПЛЕКСНА МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:

«Метод та засіб потокового шифрування на основі квазігруп. Частина 1. Генератор псевдовипадкових чисел на основі операцій з квазігрупами»

Виконав: студент 2 курсу групи 1БС-24м
спеціальності 125 Кібербезпека та захист інформації
_____ Богдан МИКИТЧЕНКО
Керівник: к. ф.-м. н., доцент каф. ЗІ
_____ Галина ШЕЛЕПАЛО
«16» грудня 2025 р.
Рецензент: к. т. н., доц., доц. каф. ПЗ
_____ Олександр ХОШАБА
«16» грудня 2025 р.

Допущено до захисту
В. о. зав. каф. ЗІ
д. т. н., проф.
_____ Володимир ЛУЖЕЦЬКИЙ
«16» грудня 2025 р.

Вінниця ВНТУ – 2025 року

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра захисту інформації
Рівень вищої освіти II (магістерський)
Галузь знань – 12 Інформаційні технології
Спеціальність – 125 Кібербезпека та захист інформації
Освітньо-професійна програма – Безпека інформаційних і комунікаційних систем

ЗАТВЕРДЖУЮ
В. о. зав. кафедри ЗІ, д. т. н., проф.
_____ Володимир ЛУЖЕЦЬКИЙ
«24» 09 2025 року

З А В Д А Н Н Я
НА КОМПЛЕКСНУ МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Микитченку Богдану Валентиновичу

1. Тема роботи: «Метод та засіб потокового шифрування на основі квазігруп. Частина 1. Генератор псевдовипадкових чисел на основі операцій з квазігрупами «,
керівник роботи: Шелепало Галина Василівна к.ф.-м.н., доцент, затверджені наказом ректора ВНТУ від 24 вересня 2025 року №313.
2. Строк подання студентом роботи 16 грудня 2025 р.
3. Вихідні дані до роботи: – генератори псевдовипадкових чисел; – алгебричний аналіз властивостей квазігруп; – програмна реалізація методів побудови квазігруп великих порядків; – експериментальне дослідження статистичної безпеки за стандартами NIST SP 800-22.
4. Зміст текстової частини: Вступ. 1 Аналіз та огляд інформаційних джерел. 2 Алгебричні властивості квазігруп для генераторів псевдовипадкових послідовностей. 3 Синтез апаратної оптимізації генератора на основі квазігруп. 4. Економічна частина. Висновки. Список використаних джерел. Додатки.
5. Перелік ілюстративного матеріалу: Зведені результати аналізу графа станів для просторів N^2 та N^3. Приклади розподілу відвідувань на торі. Патерни розподілу для масштабованих послідовностей. Порівняння методів. Візуалізація траєкторій на торі для латинських квадратів. Візуалізація розподілу відвідувань на торі для генератора. Відтворені результати E-перетворень на торі. ЛК8 від 2×4. ЛК8 від 4×2. ЛК16 з 4×4. Порівняльні характеристики квазігруп. Залежність періоду генератора від апаратної складності. Порівняння усереднених показників P-value для основних конфігурацій. Порівняння проходження тестів NIST для різних генераторів

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Галина ШЕЛЕПАЛО, к. ф.-м. н., доц. каф. ЗІ	25.09.25	26.09.25
2	Галина ШЕЛЕПАЛО, к. ф.-м. н., доц. каф. ЗІ	25.09.25	07.10.25
3	Галина ШЕЛЕПАЛО, к. ф.-м. н., доц. каф. ЗІ	25.09.25	17.10.25
4	Олександр ЛЕСЬКО, к. е. н., проф. зав. каф. ЕПВМ	25.09.25	29.10.25

7. Дата видачі завдання: 25 вересня 2025 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської дипломної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз завдання. Вступ	24.09.25 – 26.09.25	
2	Аналіз інформаційних джерел за напрямком комплексної магістерської дипломної роботи	27.09.25 – 07.10.25	
3	Розробка моделей та алгоритмів	23.10.25 – 02.11.25	
4	Практична реалізація, моделювання, результати	03.11.25 – 17.11.25	
5	Розробка розділу економічного обгрунтування доцільності розробки	18.11.25 – 22.11.25	
6	Оформлення пояснювальної записки	23.11.25 – 29.11.25	
7	Попередній захист та доопрацювання МКР	29.11.25 – 11.12.25	
8	Перевірка на наявність текстових запозичень	12.12.25 – 15.12.25	
9	Представлення МКР до захисту, рецензування	16.12.25 – 19.12.25	
10	Захист МКР	19.12.25 – 23.12.25	

Студент _____ Богдан МИКИТЧЕНКО

Керівник роботи _____ Галина ШЕЛЕПАЛ

АНОТАЦІЯ

Микитченко Б. В. Метод та засіб потокового шифрування на основі квазігруп. Частина 1. Генератор псевдовипадкових чисел на основі операцій з квазігрупами. Магістерська кваліфікаційна робота зі спеціальності 125 – Кібербезпека та захист інформації, освітньо-професійна програма – Безпека інформаційних і комунікаційних систем. Вінниця: ВНТУ, 2025. 90 с.

Укр. мовою. Бібліогр.: 30 назв; рис.: 14; табл.: 18.

Магістерська кваліфікаційна робота присвячена розв’язанню задачі підвищення криптографічної стійкості потокових шифрів шляхом розробки вдосконаленого генератора псевдовипадкових чисел (ГПВЧ) на основі алгебричних структур, а саме квазігруп. У роботі здійснено аналіз недоліків класичних рекурсивних методів генерації на основі латинських квадратів. Виявлено та теоретично обґрунтовано залежність між порядком квазігрупи (простого і складеного числа) та зв’язністю графа станів генератора. Підібрано метод побудови квазігруп великих порядків за структурою іізотопних ланцюгів в поєднанні з схрещеним доубтком. Розроблено та програмно реалізовано алгоритм побудови криптографічно стійких квазігруп великих порядків (N=256) з використанням методу «ізотопних ланцюгів» та схрещеного добутку. Цей дозволело досягти покриття простору станів графа квазігрупи понад 98%. Проведено комплексне тестування розроблених генераторів за допомогою статистичного пакета NIST SP 800-22, що підтверджено високу якість генерованих послідовностей (успішне проходження тестів на ентропію та лінійну складність). Здійснено синтез логічної структури генератора та оцінку його апаратної складності для реалізації на FPGA. В економічному розділі проведено функціонально-вартісний аналіз та обґрунтовано доцільність розробки.

Ключові слова: квазігрупи, потокове шифрування, ГПВЧ, схрещений добуток, графи станів, NIST-тести, ізотопія, криптоаналіз.

ABSTRACT

Mykitchenko B. V. Method and means of stream encryption based on quasi-groups. Part 1. Pseudorandom number generators based on operations with quasi-groups. Master's thesis in the field of 125 – Cybersecurity and Information Protection, educational and professional program – Security of Information and Communication Systems. Vinnytsia: VNTU, 2025. 90 p.

In Ukrainian. Bibliography: 30 titles; figures: 14; tables: 18.

The master's thesis is devoted to solving the problem of improving the cryptographic resistance of stream ciphers by developing an improved pseudorandom number generator (PRNG) based on algebraic structures, namely quasi-groups. The work analyzes the shortcomings of classical recursive generation methods based on Latin squares. The dependence between the order of the quasi-group (simple and composite number) and the connectivity of the generator state graph has been revealed and theoretically substantiated. A method for constructing large-order quasi-groups based on the structure of isotope chains in combination with cross-doubling has been selected. An algorithm for constructing cryptographically resistant quasi-groups of large orders (N=256) using the “isotope chains” method and cross-product has been developed and implemented in software. This has made it possible to achieve more than 98% coverage of the state space of the quasi-group graph. Comprehensive testing of the developed generators was carried out using the NIST SP 800-22 statistical package, which confirmed the high quality of the generated sequences (successful passing of entropy and linear complexity tests). The logical structure of the generator was synthesized and its hardware complexity for implementation on FPGA was evaluated. In the economic section, a functional-cost analysis was performed and the feasibility of the development was justified.

Keywords: quasi-groups, stream cipher, GPCH, cross product, state graphs, NIST tests, isotopy, cryptanalysis.

ЗМІСТ

ВСТУП

Актуальність теми. Забезпечення конфіденційності даних у сучасних комунікаційних системах вимагає використання швидкодіючих потокових шифрів, стійких до алгебраїчних та кореляційних атак. Класичні генератори на основі регістрів зсуву (LFSR/NLFSR) вичерпують свій потенціал через компроміс між лінійною складністю та швидкодією. Перспективним напрямом є використання неасоціативних алгебричних структур — квазігруп, що дозволяють будувати ефективні криптографічні примітиви.

Вагомий внесок у дослідження квазігруп у криптографії зробили закордонні вчені С. Марковський, Д. Глігороські, В. Щербаков. В Україні розвитком цього напрямку займаються науковці ВНТУ, зокрема професор В. А. Лужецький та доцент Г. В. Шелепало. Попри досягнення, існуючі методи генерації мають проблеми з фрагментацією простору станів та складністю масштабування, що зумовлює актуальність розробки нових методів синтезу квазігруп для криптографічних застосувань.

Зв'язок роботи з науковими програмами. Робота виконана згідно з планами НДР кафедри захисту інформації ВНТУ за напрямком розробки перспективних засобів криптографічного захисту.

Мета і завдання роботи. Метою є підвищення криптографічної стійкості генераторів псевдовипадкових послідовностей (ГПВП) шляхом розробки методів синтезу квазігруп великих порядків та оптимізації архітектури генератора.

Для досягнення мети вирішено такі завдання:

1. Проаналізувати методи побудови потокових шифрів на алгебричних структурах.
2. Дослідити топологію графів станів генераторів та встановити умови досягнення максимального періоду.
3. Розробити метод синтезу квазігруп порядку N=256 на основі схрещеного добутку з використанням ізотопії.

4. Здійснити апаратну оптимізацію логічної структури генератора для FPGA.
5. Провести статистичне тестування (NIST SP 800-22) та порівняльний аналіз розроблених засобів.

Об’єкт дослідження – процес генерації псевдовипадкових послідовностей для потокового шифрування.

Предмет дослідження – методи та засоби синтезу ГПВП на основі квазігруп.

У роботі використано такі методи дослідження: *методи загальної алгебри* (властивості квазігруп, ізотопія); *теорію графів* (аналіз простору станів); *статистичний аналіз* (оцінка якості за NIST); *комп’ютерне моделювання* (верифікація алгоритмів); *синтез цифрових схем* (мінімізація булевих функцій).

Науковою новизна. У роботі вперше сформульовано «Гіпотезу залежності зв'язності від простоти порядку», яка доводить, що квазігрупи простих порядків (N>3) формують єдиний граф станів, забезпечуючи максимальний період генерації. Удосконалено метод побудови квазігруп великих порядків (N=256) через введення каскадних ізотопних перетворень, що збільшило покриття простору станів з 50% до 98 %. Набуло подальшого розвитку розуміння зв’язку між періодом та криптостійкістю: доведено необхідність використання композитних архітектур замість простої рекурсії.

Апробація результатів. Основні положення доповідалися на LIII науково-технічній конференції *SMICS: Безпека сучасних інформаційно-комунікаційних систем: матеріали міжнар. наук.-техн. конф., м. Львів, 16-18 жовтня 2025 р.* ЛНУ ім. І. Франка, 2025.

1 АНАЛІЗ ТА ОГЛЯД ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1.1. Потокові шифри: архітектура, вимоги, роль ГПВЧ

Потокові шифри є фундаментальним класом симетричних криптографічних алгоритмів, що оперують на малих одиницях даних, зазвичай побітово або побайтово. На відміну від блокових шифрів, потокові шифри генерують ключовий потік, який поєднується з відкритим текстом, найчастіше за допомогою операції XOR.

У домінуючій архітектурі, відомій як синхронні потокові шифри, ключовий потік генерується абсолютно незалежно від відкритого тексту та шифротексту. Процес шифрування та дешифрування є ідентичним. Тому, серцем синхронного потокового шифру є Генератор Псевдовипадкових Послідовностей (ГПВЧ), який називають генератором ключового потоку.

Завдання ГПВЧ — взяти короткі, фіксовані входи (секретний ключ K унікальний для кожного повідомлення вектор ініціалізації або нонс) і детерміністично «розтягнути» їх у довгий ключовий потік S. Вся криптографічна безпека системи покладається виключно на якість цього потоку. Якщо зловмисник може будь-яким чином передбачити, відновити або відрізнити S від випадкової послідовності, шифр вважається зламаним.

Вимоги до такого ГПВЧ є надзвичайно суворими. А саме, він повинен продукувати послідовності з величезним періодом, оскільки будь-яке повторне використання ключового потоку миттєво призводить до катастрофічної «атаки двох блокнотів» (two-time pad attack). Вихідна послідовність має бути статистично невідрізною від істинного шуму, успішно проходячи стандартизовані набори тестів, NIST SP 800-22 [1].

Проте гарної статистики недостатньо. Потік має бути криптографічно непередбачуваним. Тобто, знаючи будь-яку частину потоку, має бути обчислювально неможливо передбачити наступні або відновити попередні біти. Ця властивість забезпечується високою лінійною складністю для протидії атаці Берлекемпа-Мессі та стійкістю до складніших атак, зокрема кореляційні та

алгебричні атаки [2, 3]. Сучасні ГПВЧ (ChaCha20) використовують нелінійні операції (додавання, ротація, XOR) для швидкого досягнення цих властивостей [4]. Безпечна схема ініціалізації ключем та нонсом є критичною: використання одного ключа з різними нонсами повинно гарантовано породжувати два статистично незалежні ключові потоки [5]. Отже, розробка ГПВЧ є складним завданням, що балансує між теоретичною безпекою, доведеною стійкістю та високою продуктивністю.

1.2. Огляд класичних ГПВЧ: LFSR, NLFSR

Будівельними блоками для більшості апаратних ГПВЧ є регістри зсуву з лінійним зворотним зв'язком (LFSR, Linear Feedback Shift Register). Їх широка популярність була зумовлена винятковою простотою апаратної реалізації, високою швидкістю роботи та глибоко розвиненою математичною теорією, що базується на поліномах над полем Галуа GF(2) [6]. При правильному виборі твірного полінома (зворотного зв'язку), *n*-бітний LFSR здатен генерувати послідовність максимального періоду 2^n-1.

Проте, ця ж математична простота та структурованість є фатальною криптографічною слабкістю. Лінійність означає, що кожен біт вихідної послідовності є простою XOR-сумою певних бітів внутрішнього стану. Це призводить до катастрофічно низької лінійної складності. Основною атакою на такі генератори є алгоритм Берлекемпа-Мессі (Berlekamp-Massey). Маючи у своєму розпорядженні лише $2n$ послідовних бітів вихідного потоку, зловмисник може за поліноміальний час повністю відновити лінійну рекурсію (тобто, твірний поліном) і згенерувати всю послідовність у майбутньому і в минулому [6]. З цієї причини LFSR у чистому вигляді зараз не використовуються, а лише є компонентами у складніших конструкціях (наприклад, у комбінаційних генераторах або генераторах з фільтрацією).

Як пряма відповідь на загрозу лінійності, були розроблені регістри зсуву з нелінійним зворотним зв'язком (NLFSR, Non-Linear Feedback Shift Register). Їх наступний стан обчислюється за допомогою нелінійної булевої функції f від бітів

поточного стану. Це рішення ефективно руйнує лінійну структуру і робить генератор стійким до атаки Берлекемпа-Мессі, дозволяючи досягти високої лінійної складності [7].

Проте NLFSR привнесли власний набір теоретичних та практичних проблем. На відміну від LFSR, відсутня проста загальна теорія для гарантування максимального періоду послідовності; дизайн NLFSR з довгим періодом є нетривіальною задачею. Вони стали мішенню для більш потужних класів атак. Алгебричні атаки є небезпечними: вони моделюють роботу генератора як систему поліноміальних рівнянь над GF(2) і намагаються розв'язати її (наприклад, методами лінеаризації або з використанням базисів Гребнера), щоб відновити внутрішній стан [8]. Актуальними є кореляційні атаки, якщо нелінійна функція f погано спроєктована і має високу кореляцію з якоюсь лінійною функцією [6, 8].

Отже, класичні підходи демонструють фундаментальний компроміс: LFSR є швидкими, простими і мають гарантований період, але є лінійними і небезпечними; NLFSR вирішують проблему лінійності, але є складнішими в аналізі періоду та вразливими до алгебричних атак. Це обґрунтовує необхідність пошуку нових алгебричних структур для побудови ГПВЧ, які б за своєю природою були нелінійними, але водночас мали достатньо багату структуру для аналізу та гарантування криптографічних властивостей.

1.3. Алгебричні основи криптографії

Сучасна криптографія нерозривно пов'язана з алгебричними структурами. В основі багатьох доведених алгоритмів, від AES до криптографії на еліптичних кривих, лежать групи та скінченні поля, зокрема поля Галуа GF. Ці структури забезпечують чітко визначені операції з властивостями асоціативності, комутативності та оборотності. Проте, ця структурованість, а саме лінійність у полях GF(2), може бути використана в алгебричних атаках [9]. Це спонукає до пошуку менш «регулярних» структур, які б зберігали ці властивості, але додавали нелінійність.

У цьому контексті ключову роль відіграють квазігрупи. Квазігрупа — це множина Q з бінарною операцією (*), для якої система рівнянь $a * x = b$ та $y * a = b$ завжди має єдині розв'язки x та y для будь-яких елементів a, b з множини Q [10]. У скінченному випадку це означає, що таблиця Келі (таблиця множення) квазігрупи є латинським квадратом. Отже, перехід від термінології «латинських квадратів» до «квазігруп» є кроком від комбінаторного опису (таблиці) до алгебричного аналізу (розв’язування системи рівнянь) та математичного аналізу (властивостей оборотності).

Для криптографії квазігрупи мають дві фундаментальні переваги: 1) Оборотність операцій: означення квазігрупи гарантує, що операція (*) є оборотною. Якщо $z = x * y$, то існують унікальні «ліве» (\) та «праве» (/) ділення, такі що $y = x \setminus z$ та $x = z / y$. Це критично важливо для шифрування, оскільки забезпечує можливість дешифрування, а в контексті ГПВЧ дозволяє аналізувати відновлення стану. 2) Неасоціативність: на відміну від груп, квазігрупа неасоціативна $(x * y) * z \neq x * (y * z)$. Саме ця властивість є джерелом нелінійності. При побудові рекурсивних генераторів, подібних до NLFSR, використання неасоціативної операції руйнує прості лінійні та алгебричні залежності, які є основою для класичних атак на LFSR та алгебричних атак на NLFSR [11]. Отже, квазігрупи являють собою ідеальний компроміс: вони зберігають необхідну для криптографії оборотність, але відкидають асоціативність, вносячи складність та нелінійність. Це дозволяє проєктувати ГПВЧ, які потенційно стійкі до стандартних атак, при цьому спираючись на багату алгебричну теорію для аналізу їхніх властивостей, таких як довжина періоду, розподіл циклів та алгебричну складність [10].

1.4. Огляд використання квазігруп у криптографії

Використання квазігруп у криптографії зосереджено у двох напрямках: розробка ГПВЧ та побудова хеш-функцій.

У сфері ГПВЧ, відомі роботи Марковського, Глігороського та ін., де пропонують неалгебричну, а емпіричну класифікацію квазігруп, а саме аналіз

статистичних властивостей послідовностей, згенерованих ітеративними квазігруповими перетвореннями, за допомогою візуального тесту «випадкового прогулянка по тору», що дозволило розділити квазігрупи на «лінійні» (які генерують структуровані, невипадкові послідовності і є поганими для криптографії) та «експоненційні» (які демонструють хаотичну поведінку та є добрими кандидатами для ГПВЧ) [12]. На основі цих «хороших» квазігруп пізніше були запропоновані конкретні архітектури потокових шифрів [13].

Найвідомішою хеш-функцією є Edon80 [14], з конкурсу SHA-3. Її архітектура базується на 8-бітних квазігрупах (порядку 256) і використовує їхні властивості для забезпечення поширення та нелінійного перемішування. Розробники стверджували, що складна і неасоціативна алгебрична структура зробить її стійкою до диференційного та лінійного криптоаналізу.

Проте, саме на прикладі Edon80 став очевидний і критичний недолік квазігрупового підходу. Складна алгебрична структура не гарантує криптографічної стійкості. У 2010 році було опубліковано низку робіт, що продемонстрували ефективні атаки на Edon80, це дозволило знаходити колізії значно швидше, ніж повним перебором [15]. Якщо операцію квазігрупи можна описати системою відносно простих поліноміальних рівнянь над полем GF(2), то весь генератор або шифр може бути зламаний методами розв'язання таких систем [16]. Отже, при проєктуванні власного ГПВЧ недостатньо взяти будь-яку квазігрупу, а ретельно проаналізувати її на приховані алгебричні структури.

1.5. Аналіз попередніх результатів дослідження

Попередні дослідження [17, 18] є емпіричним фундаментом для даної роботи, де запропоновано та програмно реалізовано рекурсивний метод побудови ГПВП, що базувався на комбінаторному об'єкті «латинський квадрат» (ЛК). Зокрема, встановлено, що запропоновані конструкції (рекурсія на двох ЛК) здатні генерувати послідовності, що успішно проходять ключові статистичні тести з набору NIST SP 800-22. Та попередня оцінка апаратної складності (~71 GE) показала потенціал методу для застосування у полегшеній криптографії [18].

Водночас, емпіричний підхід виявив фундаментальні прогалини, що неможливо пояснити, залишаючись у рамках комбінаторної теорії ЛК:

– Аномалії періоду: висунута гіпотеза про максимальний період ($T = n^{N+1}$) була спростована експериментом на ЛК 7-го порядку.
– Феномен «Нульових початкових значень» (НПЗ): виявлено існування відокремлених, коротких циклів, що слугує, в даній роботі, для виявлення неоднорідної структури графа станів генератора.

Ці результати чітко демонструють, що властивості генератора (період, структура циклів) визначаються не просто комбінаторним заповненням таблиці, а глибшими алгебричними властивостями операції, яку цей ЛК описує. Попередні роботи [17, 18] виявили наслідки, але не змогли пояснити їх причини.

1.6. Постановка задачі дослідження

Виявлені аномалії з аналізу попередніх робіт: помилка в гіпотезі періоду, феномен НПЗ доводять що комбінаторного підходу (ЛК) недостатньо і необхідний перехід до глибшого алгебричного аналізу (квазігруп) та матиматичного аналізу (оборотність). Крім того, попередні експерименти обмежувалися генерацією коротких послідовностей, які не пройшли повний цикл тестування за стандартами NIST, вимагаючи на вході мегабайти даних.

Метою даної магістерської роботи є алгебричний, статистичний, криптографічний аналіз вдосконаленого класу ГПВЧ на основі квазігруп, здатного генерувати довгі, криптографічно стійкі послідовності. Для досягнення цієї мети необхідно розв’язати такі задачі:

1) Теоретична верифікація: провести формальний алгебричний аналіз рекурсивного методу для теоретичного підтвердження або спростування гіпотез, висунутих у [17, 18]. Знайти алгебричне пояснення аномалії періоду та природи «Нульових початкових значень».

2) Розробка методу масштабування: розробити та оптимізувати алгоритми синтезу квазігруп високих порядків (до N=256) на основі методу схрещеного

добутку та каскадних схем. Дослідити вплив ізотопних перетворень на руйнування структурних симетрій при масштабуванні.

3) Емпірична класифікація та тестування: здійснити порівняльний аналіз псевдовипадкових властивостей генераторів різних архітектур (проста рекурсія, подвійна рекурсія, E-перетворення) із використанням методу «випадкового прогулянка по тору» та статистичних критеріїв.

4) Апаратний синтез та оптимізація: дослідити ефективність переходу від табличного представлення квазігруп до логічних функцій (поліномів Жегалкіна). Встановити взаємозв’язок між алгебричною складністю булевих функцій, апаратною вартістю реалізації та періодичними властивостями генератора.

5) Формулювання критеріїв проєктування: на основі отриманих експериментальних даних синтезувати набір вимог до параметрів ГПВЧ, що гарантують цілісність простору станів та високу криптографічну стійкість.

2 АЛГЕБРИЧНІ ВЛАСТИВОСТІ КВАЗІГРУП ДЛЯ ГЕНЕРАТОРІВ ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ

У даному розділі обґрунтовано та розроблено методологію дослідження генераторів псевдовипадкових послідовностей (ГПВП) на основі квазігруп через аналіз їхніх графів станів. Замість традиційного комбінаторного підходу до латинських квадратів, запропоновано розглядати процес генерації як функціонування скінченного автомата, що дозволяє виявити фундаментальні зв’язки між алгебричною структурою квазігрупи та періодичними властивостями вихідної послідовності. Проведено класифікацію топологій графів станів для квазігруп різних порядків, сформульовано гіпотезу про вплив простоти порядку N на зв’язність графа та досліджено ефекти масштабування системи у простір N^3. Окрему увагу приділено емпіричній верифікації псевдовипадкових властивостей згенерованих послідовностей за допомогою візуальних тестів.

2.1 Розробка програмного комплексу для аналізу графів станів

Для алгебричного аналізу було створено спеціалізований програмний комплекс, що дозволяє моделювати роботу рекурсивних генераторів та будувати їхні графи станів. Це забезпечує автоматизовану побудову орієнтованих графів для довільних квазігруп, ідентифікацію компонентів сильної зв’язності та розрахунок їх параметрів. Використання графового підходу дозволяє перейти від спостереження за окремими згенерованими послідовностями до цілісного аналізу всього простору станів генератора, виявляючи структурні особливості, приховані при стандартному тестуванні.

2.1.1. Постановка задачі експерименту

Попередні дослідження [18] виявили такі емпіричні аномалії: наявність ізольованих початкових станів та нелінійна поведінка періоду.

Існування специфічних вхідних наборів даних (раніше визначених як «Нульові Початкові Значення»), що призводять до генерації вироджених коротких послідовностей, які не належать до основного циклу генератора.

Емпіричні дані щодо максимальної довжини послідовності не корелювали з простими степеневими залежностями, що вказувало на наявність складної внутрішньої структури перетворень.

Для відповіді на ці питання перейшли від простої генерації послідовностей до глибокого аналізу простору станів генератора. Для базової рекурсії

$$\boldsymbol{u_i = Q(u_{i-1}, u_{i-2})} \qquad (2.1)$$

де Q — квазігрупа порядку N, простір станів складається з N^2 унікальних пар (u_{i-1}, u_{i-2}). Сама рекурсія моделюється як орієнтований граф (скінченний автомат), де кожна з N^2 пар є вершиною, а ребро визначає перехід від стану (a, b) до стану $(b, Q(a, b))$.

Розроблений програмний стенд (реалізований мовою Python) дозволяє для довільної квазігрупи Q будувати відповідний граф та аналізувати його топологію, зокрема знаходити всі сильно зв'язні компоненти та їхні розміри.

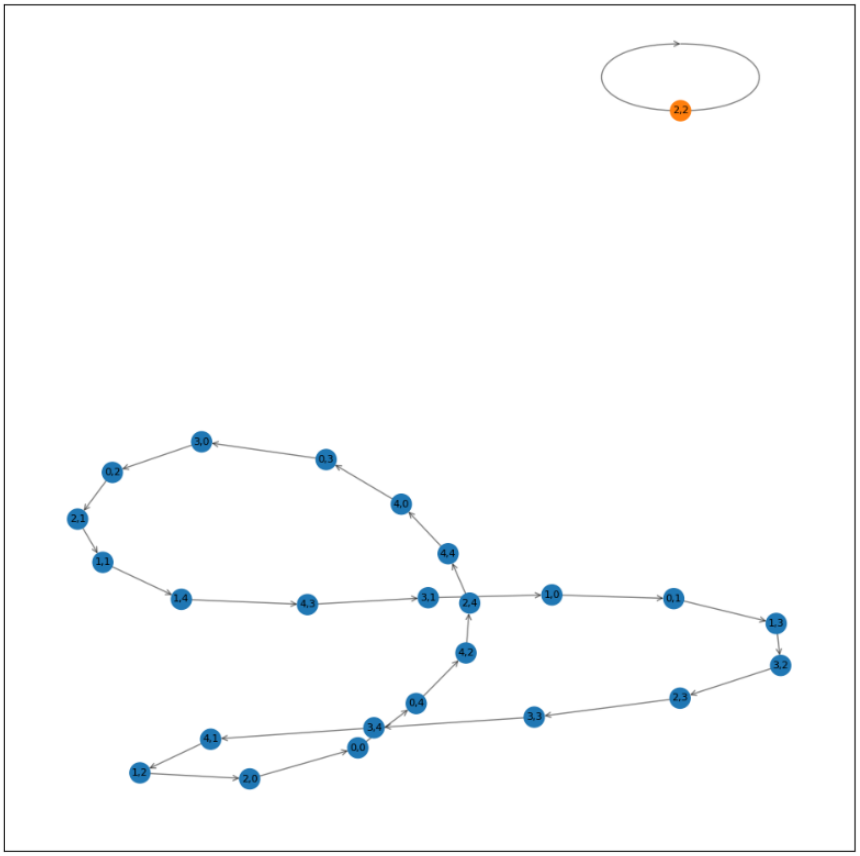


Рис 2.1 - Граф з структурою [24, 1] для ЛК порядку N=5.

Аналіз отриманих графів дозволив сформулювати ключове співвідношення для оцінки характеристик генератора:

$$\boldsymbol{L_{max} = S_{comp} + 1} \quad (2.2)$$

де L_{max} — максимальна довжина псевдовипадкової послідовності, а S_{comp}— розмір найбільшого компонента зв'язності графа. Дана формула пояснює природу ізольованих початкових пар як вершин графа, що належать до окремих компонентів малого розміру (наприклад, циклів довжиною 1 або 4) і не мають вхідних ребер з основного компонента зв'язності. Це підтверджує тезу, що саме топологічна структура графа станів є визначальною характеристикою генератора, яка диктує його поведінку.

2.1.2 Експериментальний аналіз та класифікація структур

На основі розробленого інструментарію проведено серію експериментів для квазігруп різних порядків, що дозволило класифікувати їх за топологічними ознаками графа станів. Аналіз малих порядків:

1) Для N=3 (повний перебір 12 ЛК) виявлено поділ на структури, що забезпечують покриття N^2 (цикл довжиною 9) та структури з меншим періодом.

2) Для N=4 (повний перебір 576 ЛК) зафіксовано високу варіативність топологій. Більшість квазігруп формують сильно фрагментовані графи (наприклад, <8, 8> або <3, 3, 3, 3, 3, 1>). Найкращі зразки (надалі — *оптимальні структури*) демонстрували топологію <N^2-1, 1>, що забезпечує максимальну довжину послідовності 16 (N^2). На цьому етапі було зроблено проміжний висновок, що граничною структурою є <N^2-1, 1>, а максимальною довжиною — N^2.

3) Для N=5 (виявлення абсолютно оптимальних структур) виявлено фундаментально новий клас квазігруп, відсутній у порядках N=3 та N=4, а саме:

- Оптимальні структури (довжина 26 = N^2+1): виявлено ЛК (наприклад, #68, #85, #702), які генерують єдиний компонент зв'язності розміром 25.
- Субоптимальні структури (довжина 25 = N^2): знайдено структури з топологією <24, 1>, аналогічні до тих, що спостерігалися при N=4.

Маючи робочий інструмент, було проведено серію масштабних експериментів для квазігруп різних порядків.

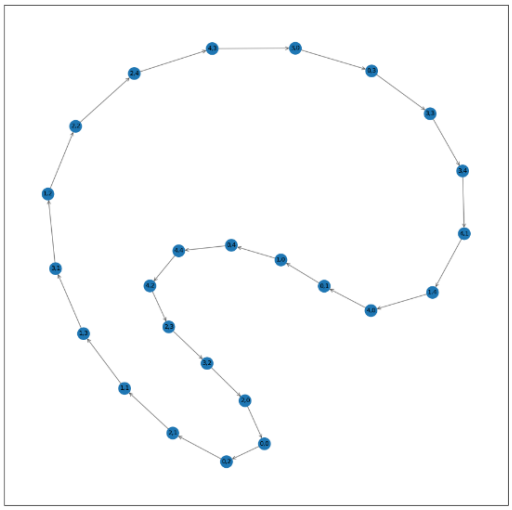


Рис 2.2 – Граф з структурою N^2+1 випадкового ЛК порядку $N=5$

Отримані дані поставили питання щодо причини відмінності у поведінці простого порядку $N=5$ та складеного $N=4$, а також простого $N=3$ без ефекту N^2+1.

На основі порівняльного аналізу висунуто Гіпотезу залежності зв'язності від простоти порядку: здатність генерувати єдиний компонент $<N^2>$ і досягати довжини N^2+1 є унікальною властивістю квазігруп простого порядку $N > 3$. Для перевірки гіпотези проведено додаткові дослідження на більших порядках:

- $N=6$: гіпотеза підтверджена для складених чисел. Ефект N^2+1 відсутній. Максимум — $N^2 = 36$, структура <35, 1>.
- $N=7$: гіпотеза підтверджена для простих чисел. Знайдено оптимальні структури з довжиною $N^2+1 = 50$ та єдиним компонентом <49>.
- $N=8$: гіпотеза підтверджена для складених чисел. Максимум — $N^2 = 64$, структура [63, 1].
- $N=11$: гіпотеза підтверджена для простих чисел. Знайдено структури з довжиною $N^2+1 = 122$ та топологією <121>.

Це формує фінальну класифікацію, наведену в таблиці 2.1.

Таблиця 2.1. Підсумкові результати аналізу графа станів ГПВЧ залежно від порядку квазігрупи N.

Порядок N	Тип числа	Макс. довжина ПВП	Оптимальна топологія графа
N=3	Просте	N^2 (9)	<8, 1>
N=4	Складене	N^2 (16)	<15, 1>
N=5	**Просте**	**N^2+1 (26)**	**<25>**
N=6	Складене	N^2 (36)	<35, 1>
N=7	**Просте**	**N^2+1 (50)**	**<49>**
N=8	Складене	N^2 (64)	<63, 1>
N=11	**Просте**	**N^2+1 (122)**	**<121>**

2.1.3 Інтерпретація результатів

Хоча в традиційній криптографії перевага здебільшого надається порядкам, що є степенями двійки (N=4, 8, 16) через зручність бінарного представлення та апаратної реалізації, отримані результати вказують на обмеженість такого підходу в контексті рекурсивних генераторів. Дослідження доводить, що використання складених порядків створює передумови для зниження стійкості системи через структурні особливості простору станів.

Важливість виявленого показника максимального періоду N^2+1 полягає не в арифметичному прирості довжини послідовності на один елемент, а в тому, що він виступає індикатором фундаментальної алгебричної властивості. Експериментально підтверджено, що для квазігруп простих порядків $N > 3$ характерна здатність утворювати єдиний сильно зв'язний граф станів, який охоплює всі N^2 вузлів без винятку.

Основним недоліком квазігруп складених порядків (N=4, 6, 8) є фрагментація графа станів. Їх алгебрична структура, що характеризується наявністю нетривіальних підструктур, призводить до розбиття простору станів на окремі компоненти. Навіть структури, що демонструють найкращі результати для цих порядків (топологія <N^2 -1, 1>), залишаються фрагментованими. Це

означає неминучу наявність ізольованих станів або їх груп, які діють як пастки і не належать до основного циклу генерації, що знижує надійність системи.

Орієнтація виключно на зручні степені двійки, зокрема N=8, підвищує ризик вибору субоптимальних параметрів. Аналіз демонструє, що випадково обрана квазігрупа порядку 8 з високою ймовірністю матиме структуру з критичним рівнем фрагментації (наприклад, набір з восьми ізольованих циклів), що робить її непридатною для криптографічних застосувань.

На основі проведеного аналізу сформульовано рекомендацію до проєктування надійних ГПВЧ на основі рекурсивного методу. Для забезпечення цілісності простору станів та уникнення ефекту ізольованих компонентів доцільно використовувати квазігрупи, порядок N яких є простим числом, більшим за 3 (наприклад, 5, 7, 11). Саме такі структури демонструють здатність до формування єдиного сильно зв'язного графа $<N^2>$, що гарантує досягнення максимального періоду $N^2 + 1$ та відсутність структурних вразливостей.

2.1.4. Аналіз графа станів як засіб попереднього відбору кандидатів

Виявлені топологічні властивості графа станів безпосередньо корелюють із статистичними характеристиками згенерованих послідовностей. Це дозволяє використовувати аналіз графа як метод попередньої селекції, що дає змогу виключити з розгляду квазігрупи з незадовільною структурою ще до етапу проведення ресурсномістких статистичних тестів.

Як приклад можна навести тест «Прогулянка на торі», який оцінює рівномірність заповнення простору станів N x N. Для успішного проходження тесту послідовність повинна мати період, достатній для відвідування всіх N^2 вузлів. Розглянута вище квазігрупа порядку N=8 із фрагментованою структурою (наприклад, набір із восьми циклів довжиною 8) гарантовано генерує послідовність із періодом T=8. Така послідовність фізично не здатна забезпечити покриття простору станів (64 елементи), що робить проведення подальшого тестування недоцільним.

Таким чином, структурний аналіз дозволяє не лише прогнозувати результати статистичних перевірок, а й інтерпретувати причини їхніх невдач через топологічні обмеження графа станів, зокрема наявність ізольованих компонентів або недостатня довжина максимального циклу.

2.2. Аналіз рекурсивних систем N^3

У параграфі 2.1 встановлено, що аналіз графа станів N^2 є вичерпним методом для класифікації квазігруп порядку N за їхніми генеруючими властивостями, і виявлено «гіпотезу простих чисел». Далі логічним кроком, з [18], є застосування цього ж методу до складнішої рекурсивної системи, яка використовує дві квазігрупи (або три попередні елементи стану):

$$u_i = ((u_{i-1} * u_{i-2}) \bullet u_{i-3}) \qquad (2.3)$$

де «*» та «•» — операції першої та другої квазігруп відповідно. Це розширює простір станів генератора до N^3 унікальних трійок.

Метою цього етапу експеримента є три завдання:

– Перевірити, чи зберігається залежність зв'язності графа станів від простоти числа N у складнішому просторі N^3.
– Дослідити вплив вибору пари квазігруп на топологію графа, зокрема проаналізувати ефективність використання двох ідентичних (ізотопних) квазігруп (пара L, L).
– З'ясувати, чи впливає порядок застосування квазігруп на результат, тобто чи є пари ($L1$, $L2$) та ($L2$, $L1$) еквівалентними за своїми властивостями.

Для отримання відповідей проведено серію експериментів з аналізу простору станів N^3 для квазігруп порядків N=3, 4, 5, 6 та 7.

2.2.1. Верифікація гіпотези у просторі N^3

В попередньому параграфі встановлено, що закономірності поведінки квазігруп простого та складеного порядку, виявлені для простору N^2, зберігаються і при переході до простору станів N^3.

Для складених чисел (N=4, 6) не виявлено жодної пари квазігруп, що забезпечувала б період N^3+1. Зокрема, для порядку N=4 було проаналізовано всі 331 776 можливих пар. Максимально досягнута довжина послідовності склала $N^3 = 64$ (виявлено 13824 пар), а топологія графа відповідала фрагментованій структурі <63, 1>. Аналогічний результат отримано для N=6 (вибірка 10000 пар): оптимальні пари досягли лише періоду $N^3 = 216$ із топологією <215, 1>.

Для простих чисел (N=5, 7) зафіксовано якісно іншу картину. У вибірці для N=5 (250 000 пар) виявлено 2081 пару квазігруп, що генерують єдиний сильно зв'язний компонент розміром $<N^3>$. Це дозволило досягти максимальної довжини послідовності $N^3+1 = 126$. Для порядку N=7 (вибірка 10000 пар) гіпотезу було підтверджено остаточно: виявлено 34 пари з оптимальною топологією <343>, що забезпечило досягнення максимальної довжини $N^3+1 = 344$. Отриманий результат для N=7 дає теоретичне обґрунтування емпіричним даним, наведеним у роботі [18]. Значення періоду 344, яке раніше розглядалося як аномалія, фактично є наслідком використання оптимальної пари квазігруп простого порядку, що генерує єдиний нерозривний граф станів.

Аналіз повного перебору 144 пар для порядку N=3 показав, що цей порядок, як і в просторі N^2, демонструє властивості, характерні для складених чисел. Максимальна зафіксована довжина склала $N^3 = 27$ із фрагментованою структурою графа <26, 1>, оптимальних структур із періодом 28 не виявлено.

Результати для просторів N^2 та N^3 наведено в таблиці 2.2.

Таблиця 2.2. Зведені результати аналізу графа станів для просторів N^2 та N^3.

Простір станів	Порядок N	Тип числа	Макс. довжина ПВП	«Чемпіонська» структура графа
N^2	N=4	Складене	**N^2**	[N^2-1, 1]
N^2	N=5	**Просте > 3**	**N^2+1**	**[N^2]**
N^2	N=6	Складене	**N^2**	[N^2-1, 1]
N^2	N=7	**Просте > 3**	**N^2+1**	**[N^2]**
N^3	N=4	Складене	**N^3**	[N^3-1, 1]
N^3	N=5	**Просте > 3**	**N^3+1**	**[N^3]**
N^3	N=6	Складене	**N^3**	[N^3-1, 1]
N^3	N=7	**Просте > 3**	**N^3+1**	**[N^3]**

Отже, експерименти доводять, що здатність генерувати ідеальний, повний цикл (довжиною N^k+1) є унікальною властивістю квазігруп простого порядку N>3. Квазігрупи складеного порядку (або N=3) характеризуються фрагментацією графа станів, що унеможливлює досягнення теоретичного максимуму.

2.2.2. Вплив вибору пари квазігруп

Окрім порядку N, критично важливим фактором, що визначає топологію графа станів системи, є вибір конкретної комбінації квазігруп Q_1 та Q_2. Дослідження спрямоване на перевірку гіпотези про вплив ідентичності компонентів та порядку їх застосування.

Досліджено припущення про використання двох ідентичних квазігруп (пара L, L) у рекурсивній схемі, що призводить до накладання їхніх внутрішніх симетрій та спричиняє фрагментацію графа станів. Для порядку N = 4 проаналізовано 576 пар вигляду (L, L). Середня кількість компонентів зв'язності склала 8.04, що свідчить про високий рівень фрагментації. Найгірший результат (пара L_1, L_1) продемонстрував розбиття простору на 22 компоненти. Жодна з досліджених ідентичних пар не дозволила досягти максимального періоду 64; найкращий результат склав 60 станів із топологією <59>. Аналогічні результати отримано для порядку N = 5 на вибірці з 500 пар. Середній рівень фрагментації склав 7.46. Максимальна зафіксована довжина послідовності дорівнювала 125 (структура <124, 1>), тоді як теоретичний максимум 126 (характерний для оптимальних різнорідних пар) досягнутий не був. Експериментально встановлено, що всі виявлені оптимальні пари (13824 для N=4 та 2 081 для N=5) належать до множини різнорідних квазігруп (L_1, L_2). Використання ідентичних квазігруп є гарантовано субоптимальним рішенням, оскільки структурна інтерференція унеможливлює досягнення максимального періоду.

Окремо перевірено вплив порядку застосування квазігруп у рекурсивній формулі, тобто еквівалентність пар (Q_A, Q_B) та (Q_B, Q_A). Для порядку N = 4 було розглянуто оптимальну пару (L_1, L_{77}), яка забезпечувала максимальну довжину циклу 64 та структуру графа <63, 1>. При інверсії пари (використання комбінації

L_{77}, L_1) зафіксовано деградацію показників: граф станів розпався на 6 компонентів із топологією <15, 15, 15, 15, 3, 1>, а максимальна довжина послідовності зменшилася з 64 до 163. Отже, досліджувана рекурсивна система некомутативна. Вибір того, яка квазігрупа є «внутрішньою» (Q_1), а яка — «зовнішньою» (Q_2), має критичне значення і впливає на топологію графа станів.

2.2.3. Вплив індивідуальних топологічних властивостей компонентів

На завершальному етапі аналізу подвійної рекурсії досліджено вплив топологічних характеристик окремої квазігрупи (визначених під час аналізу простору N^2) на її ефективність при формуванні оптимальних пар у просторі N^3.

Для порядку N=4 проведено порівняльний аналіз внеску окремих квазігруп у загальну стійкість системи. Квазігрупа з ідентифікатором #1, яка в одиночних тестах демонструвала високий рівень фрагментації, у складі пари також показала низьку ефективність: середня максимальна довжина згенерованої послідовності за її участі склала лише 22.3.

Натомість ідентифіковано групу квазігруп (наприклад, #111), які демонстрували високу структурну сумісність. Такі компоненти утворювали оптимальні пари значно частіше за середньостатистичний показник (64 успішні комбінації з 576 можливих).

Отримані результати підтверджують, що індивідуальні алгебричні властивості компонентів мають визначальне значення для результуючої системи. Квазігрупи, що характеризуються оптимальною (нефрагментованою) структурою графа станів у просторі N^2, є найбільш перспективними кандидатами для побудови складних рекурсивних генераторів у просторі N^3.

2.2.4. Загальні висновки для подальших досліджень

Проведений аналіз фазового простору N^3 дозволив теоретично обґрунтувати емпіричні аномалії, виявлені у попередніх дослідженнях [18], зокрема природу періоду T=344. Результати моделювання довели, що властивості генератора визначаються не випадковим перебором, а чіткими алгебричними

закономірностями. На основі цього сформульовано фундаментальні критерії для проєктування стійких ГПВП:

1. Критерій топологічної цілісності (вплив порядку N): Встановлено, що здатність формувати єдиний, нефрагментований граф станів є унікальною властивістю квазігруп простих порядків ($N > 3$). Використання складених чисел неминуче призводить до розпаду простору станів на ізольовані компоненти.
2. Критерій структурної асиметрії: Доведено необхідність використання композиції структурно різнорідних (неізотопних) квазігруп ($L1$, $L2$). Використання ідентичних компонентів (L, L) спричиняє ефект «колапсу симетрії», що критично зменшує період генератора.
3. Критерій спадковості властивостей: Ефективність складної рекурсивної системи прямо корелює з якістю її базових компонентів. Квазігрупи, що демонструють оптимальну топологію графа у просторі N^2, є пріоритетними кандидатами для побудови систем вищих порядків.

Ці висновки змінюють парадигму подальших досліджень: від сліпого стохастичного пошуку до цілеспрямованого синтезу квазігруп із наперед заданими алгебричними властивостями, що буде реалізовано у наступних розділах.

2.3. Емпірична оцінка псевдовипадкових властивостей методом «Випадкової прогулянки по торі»

Як було зазначено в аналізі літературних джерел, алгебричні властивості квазігруп не завжди дають вичерпну інформацію про їхню криптографічну стійкість. У роботі Марковського та Глігороського [12] запропоновано метод емпіричної класифікації квазігруп, заснований на візуальному тесті «Випадкова прогулянка по тору» (Random Walk on Torus). Цей підхід дозволяє розділити квазігрупи на «лінійні» (погані) та «експоненційні» (хороші) залежно від того, наскільки хаотично згенерована послідовність покриває площину.

У рамках даного дослідження було проведено серію експериментів із застосуванням цього методу до розробленого рекурсивного генератора. Метою експерименту було перевірити, чи зберігають квазігрупи, класифіковані в літературі як «експоненційні», свої властивості при зміні алгоритму генерації.

2.3.1. Методика тестування та результати для квазігруп порядку N=4

Для експерименту створено програмний стенд, що інтерпретує елементи послідовності як вектори руху на тороїдальній сітці *M* x *M*. При *N*=4 значення 0 –3 відповідають чотирьом ортогональним напрямкам. Тестування одного ЛК показало максимальний період 15–16 елементів, що на сітці 50 x 50 не дає інформативної візуалізації розподілу (рис. 2.3).

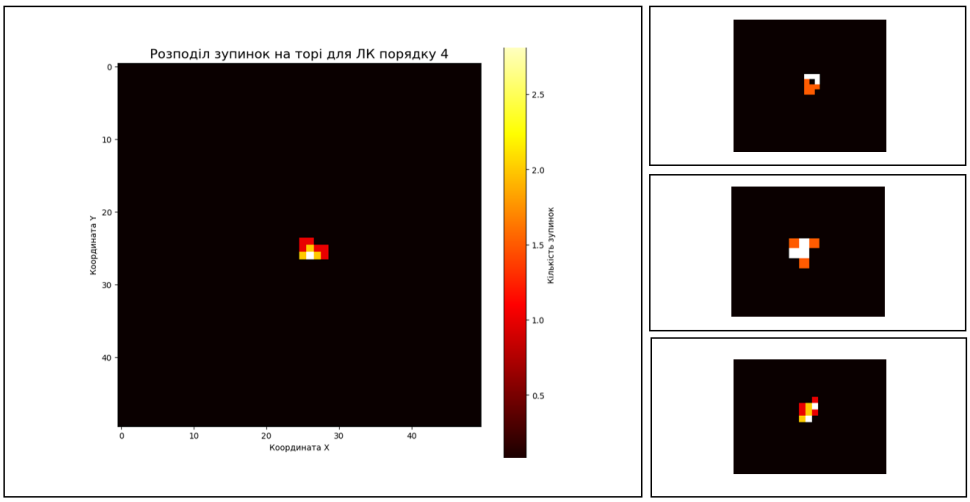


Рис. 2.3. Приклади розподілу відвідувань на торі для коротких послідовностей (*N*=4)

Для інтерпретації отриманих значень χ^2 (хі-квадрат) слід враховувати статистичну природу критерію. Він оцінює сумарне відхилення реальної кількості відвідувань кожної клітинки тора від теоретично очікуваної (у випадку ідеального рівномірного розподілу). Кількість ступенів вільності для тора розміром *M* x *M* становить $df = M^2 - 1$. Для тора 40 x 40 ($df = 1599$) критичне значення χ^2, при якому гіпотеза про випадковість розподілу не відкидається (з рівнем значущості $\alpha = 0.05$), має знаходитися в межах ≈ 1500...1700. Будь-які значення, що суттєво перевищують цей поріг (наприклад, $\chi^2 > 2000$, а тим паче значення порядку 10^6), свідчать про наявність сильних кореляцій,

закономірностей та нерівномірності покриття простору станів, що є ознакою незадовільної якості генератора.

Статистичний аналіз за критерієм x^2 показав надвисокі значення (наприклад, 4485), підтвердивши значну нерівномірність розподілу, що є очікуваним для малого періоду. Для виявлення патернів застосували циклічне дублювання послідовності (до 10000 разів). Масштабування виявило, що замість хаотичного шуму, характерного для експоненційних квазігруп, усі ЛК порядку 4 формують виражену лінійну структуру (рис. 2.4).

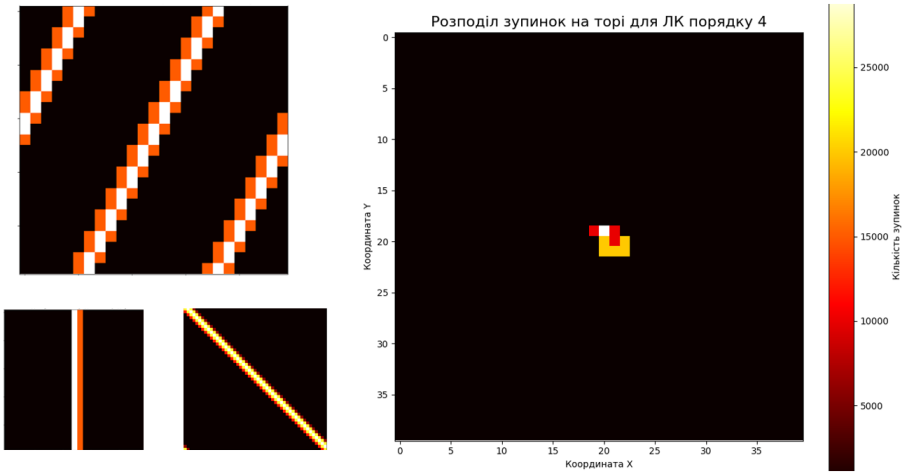


Рис. 2.4. Патерни розподілу для масштабованих послідовностей (N=4). Найбільший рисунок – аномалія.

Статистика x^2 для цих випадків залишається незадовільною, підтверджуючи, що повторення короткого циклу не покращує ентропійні властивості генератора.

Аномалію виявлено для асиметричного квадрата QG4-6 (рядок 0 1 2 3). Хоча в літературі [12] його класифікують як експоненційний із якісним шумом, рекурсивний метод дав період 16. Масштабування на торі 40x40 (10^4 повторень) замість діагоналей чи рівномірного покриття виявило жорстку локалізацію траєкторії в центрі сітки (рис. 2.4).

Статистичний показник для цього випадку сягнув екстремального значення $\chi^2 \approx 3.18 \times 10^7$. Такий результат дозволяє стверджувати, що властивість «експоненційності» (здатності генерувати шум) не є інваріантною характеристикою самої квазігрупи, а критично залежить від обраного алгоритму

генерації. Той самий алгебричний об'єкт (QG4-6), який є ефективним в одній схемі, демонструє вироджену поведінку в іншій.

2.3.3. Порівняльний аналіз властивостей ЛК у різних методах генерації

Для глибшого розуміння зв’язку між алгебричною структурою латинського квадрата та псевдовипадковими властивостями згенерованих на його основі послідовностей було проведено додаткове порівняльне дослідження. В якості об’єктів аналізу були обрані конкретні ЛК, які у роботі [12] демонстрували характерні візуальні патерни (лінійні смуги, симетрії) при використанні методу E-перетворень рядків.

Метою експерименту було перевірити, чи зберігаються ці візуальні особливості при використанні запропонованого у даній роботі рекурсивного методу ($u_i = u_{i-1} * u_{i-2}$). Для цього згенеровані послідовності були візуалізовані на торі, і отримані результати зіставлені з даними з першоджерела.

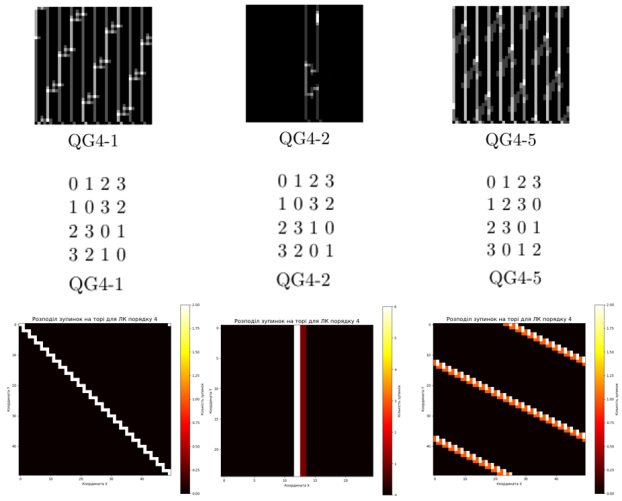


Рис. 2.5. Порівняння методів та їх результатів

Як видно з рисунку 2.5, хоча точні візерунки відрізняються, загальна тенденція до формування геометричних структур зберігається.

Для ЛК, класифікованих як «лінійні» в [12] (наприклад, QG4-1, QG4-2), рекурсивний метод також генерує послідовності з чітко вираженими діагональними смугами або регулярними ґратками.

Підтверджено, що внутрішні симетрії таблиці Келі (підквазігрупи, зсуви) проявляються у вихідній послідовності незалежно від методу генерації. Ця

фундаментальна лінійність зберігається крізь алгоритми перетворень, що робить такі структури непридатними для криптографії без додаткової обфускації.

2.3.4. Дослідження квазігруп вищих порядків

Логічним продовженням експерименту стало дослідження впливу порядку квазігрупи N на якість генерації. Було висунуто припущення, що збільшення розмірності простору станів може призвести до руйнування лінійних залежностей і появи більш хаотичних структур. Для перевірки цієї гіпотези було проведено серію тестів для ЛК порядків $N = 5, 6, ..., 20$. (Повний набір результатів наведено у Додатку Б).

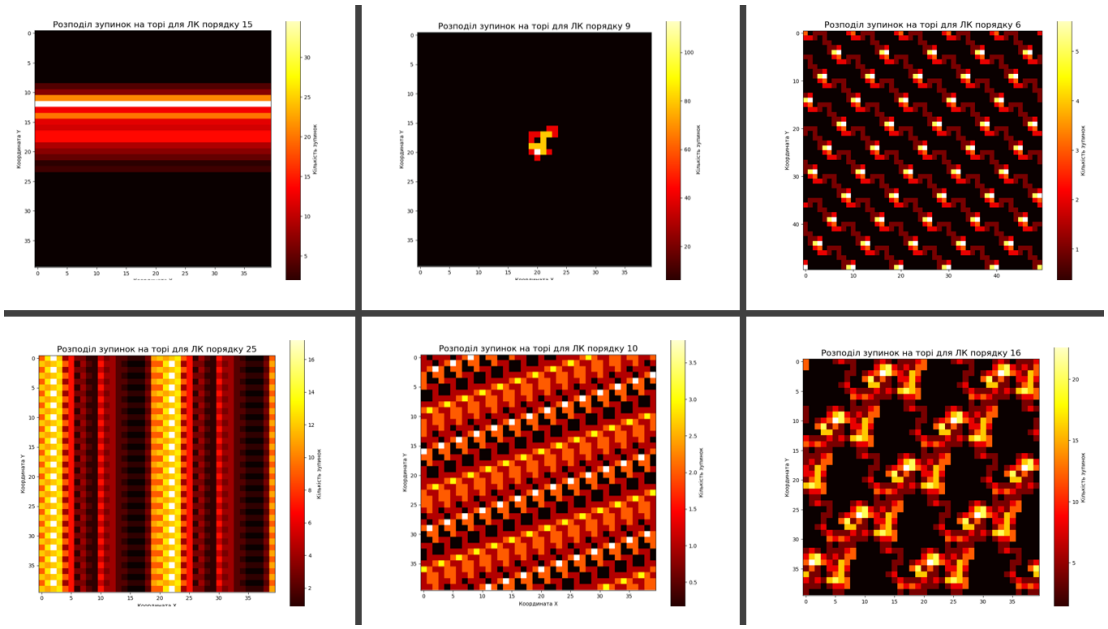


Рис. 2.6. Візуалізація траєкторій на торі для латинських квадратів різних порядків ($N = 6–25$)

Зростання N збільшує період, але не покращує статистику: замість шуму формуються впорядковані орнаменти, а критерій χ^2 вказує на нерівномірність розподілу. Доведено, що просте масштабування (N=4–20) не підвищує ентропію. В автономній рекурсії внутрішні симетрії спричиняють виродження траєкторій у лінійні патерни, що робить ізольовану квазігрупу непридатною для стійкого генератора та вимагає переходу до композитних архітектур.

2.3.5. Дослідження псевдовипадкових послідовностей на основі композиції двох ЛК

Наступним етапом стало тестування більш складної конструкції генератора, що використовує рекурсію на базі двох латинських квадратів:

$$u_i = ((L_1[u_{i-1}][u_{i-2}]) * L_2[u_{i-3}]) \quad (2.4)$$

Експерименти з парами ЛК порядку 4 довели, що введення нелінійності та збільшення періоду (до 64) не усуває фундаментальну лінійність: траєкторії зберігають геометричні патерни, а високі значення x2 підтверджують сильні кореляції. Ускладнення рекурсії не нівелює вплив внутрішньої структури на малих порядках. При $N > 6$ траєкторії стають насиченішими, але замість хаотичного шуму формують складні орнаменти (Рис. 2.7).

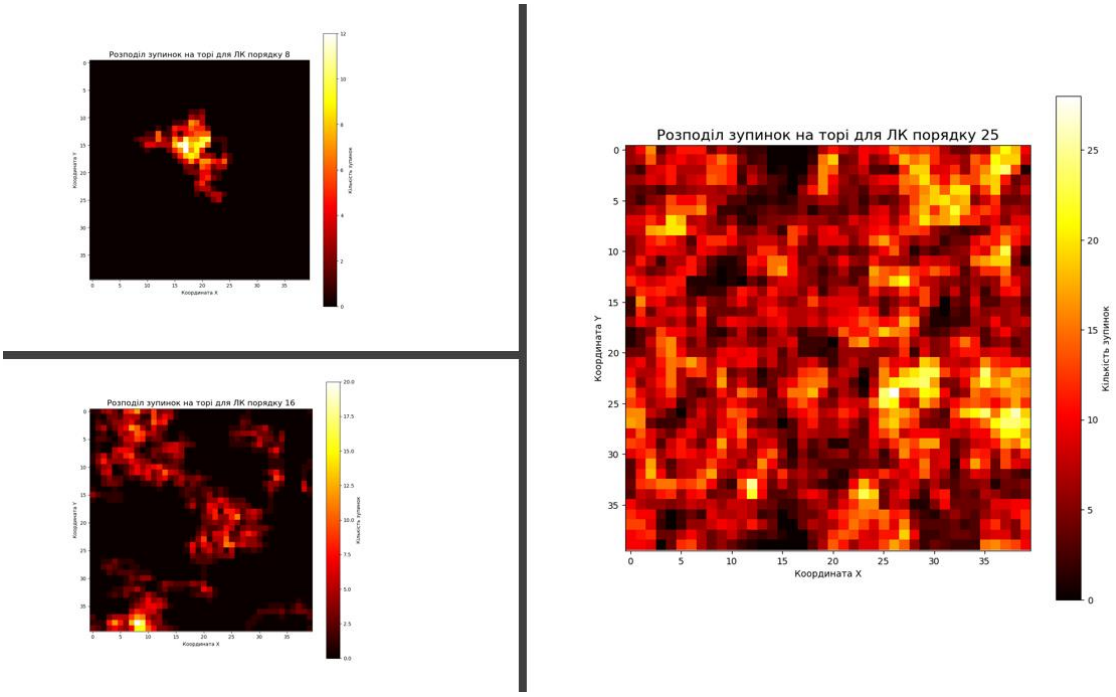


Рис. 2.7. Візуалізація розподілу відвідувань на торі для генератора на основі двох ЛК без циклічного повторення для N=8, 16, 25.

Виокремлено характерні типи поведінки: локалізація в кластерах (N=8), горизонтальні смуги через структурні залежності (N=16) та найпоширеніші діагональні ґратки (N=25). Хоча останні покривають значну площу, вони

підпорядковані суворому геометричному закону. Наявність будь-якого повторюваного візерунка підтверджує непридатність генератора для криптографічних задач.

2.3.6. Висновки до експериментального дослідження та перегляд методології

Експерименти підтвердили, що проста рекурсія зберігає структурні особливості вихідних квазігруп, формуючи на торі геометричні патерни замість шуму незалежно від зростання порядку *N*. Встановлено, що методика візуалізації, ефективна для E-трансформацій [12], є неінформативною для автономних генераторів через їхній короткий період. Для адекватної оцінки «експоненційності» необхідно використовувати квазігрупу як фільтр вхідних даних, а не як самостійний генератор, що дозволить відокремити її властивості від артефактів алгоритму.

2.4. Верифікація методу класифікації шляхом моделювання E-перетворень рядків

Результати попередніх експериментів з рекурсивним генератором виявили невідповідність між очікуваними властивостями так званих експоненційних квазігруп та їх реальною поведінкою. Для виключення можливості помилки в інструментарії тестування та для валідації самої концепції емпіричної класифікації було проведено повне відтворення експерименту, описаного в базовій літературі. Основною відмінністю цього етапу стала зміна архітектури генерації: замість автономної рекурсії було реалізовано механізм квазігрупових рядкових перетворень (E-transformations).

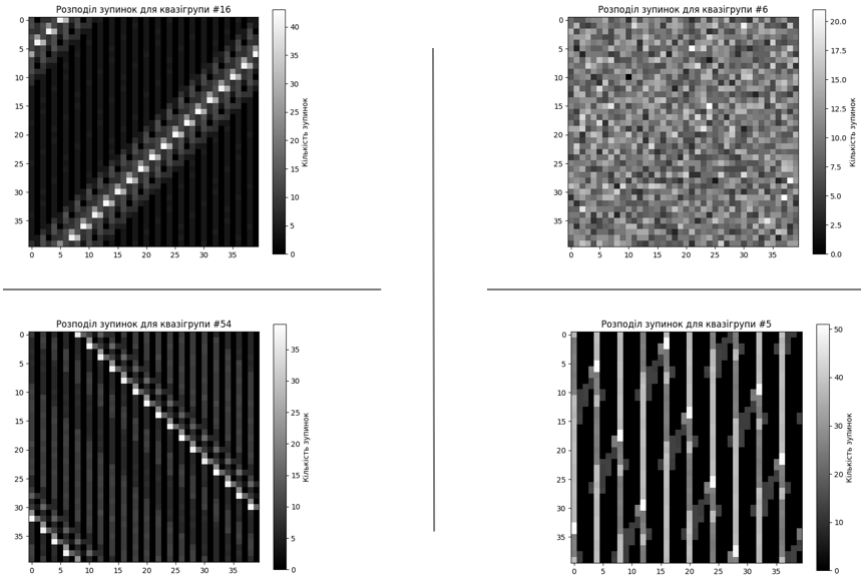


Рис. 2.8. Відтворені результати E-перетворень на торі

У даній моделі на вхід системи подається періодична послідовність, яка піддається багатократним перетворенням за законом:

$$b_i = b_i - 1 * a_i \qquad (2.5)$$

де a_i — вхідний символ, b_i — вихідний символ, а «*» позначає операцію квазігрупи. Така схема моделює проходження даних крізь ланцюг перетворювачів, що є типовим для архітектури потокових шифрів та хеш-функцій. Отримані вихідні послідовності були протестовані за методом «випадкова прогулянка по тору» з розміром площини 40 × 40 точок, що дозволяє оцінити рівень хаотичності та рівномірності розподілу траєкторій.

У рамках експерименту було розроблено програмний модуль, що реалізує послідовне застосування E-перетворень до вхідної послідовності довжиною 968 000 елементів. Тестування проводилося на вибірці квазігруп порядку N=4. Оцінка якості розподілу здійснювалася за допомогою критерію хі-квадрат (χ^2) для чотирьох рівнів деталізації розбиття тора (від 4 до 25 регіонів) з різними рівнями значущості (p=0.05 та p=0.075/0.1).

2.4.1. Аналіз отриманих результатів

Результати комп’ютерного моделювання виявили чітку поляризацію досліджуваних алгебричних структур, що повністю узгоджується з класифікацією, наведеною у базовій літературі. Отримані емпіричні дані дозволили розділити вибірку на дві діаметрально протилежні групи за характером заповнення простору станів.

Перша група квазігруп (у тестовій вибірці представлені зразками №5, №16, №54 див. рис. 2.8) продемонструвала яскраво виражену лінійну поведінку. Для цих структур спостерігалися екстремально високі значення статистики хі-квадрат, що на порядки перевищували критичні пороги. Зокрема, для тестів розбиття на 4 регіони (R1-S1) розрахункові значення сягали 8193, а для 25 регіонів (R4-S7) — понад 33900 при критичному значенні 37.6. Такі квазігрупи провалили 100% тестів (8 з 8), що свідчить про повну відсутність псевдовипадкових властивостей та наявність жорстких функціональних залежностей у згенерованій послідовності. Візуалізація траєкторій для цієї групи підтвердила формування чітких геометричних патернів замість хаотичного шуму.

Друга група квазігруп (зразок №6 див. рис. 2.8) показала значно кращі статистичні результати, які можна класифікувати як прояв псевдовипадкових властивостей. Значення критерію хі-квадрат для цих структур знаходилися значно ближче до критичних меж, а в окремих випадках (тести R1-S1, R4-S7) успішно проходили перевірку на рівномірність розподілу. Наприклад, квазігрупа №6 продемонструвала значення $\chi^2 \approx 6.7$ для 4 регіонів, що є статистично прийнятним результатом. Хоча ці квазігрупи не пройшли повний спектр тестів (провал від 1 до 4 тестів із 8), характер їхнього розподілу кардинально відрізняється від лінійних аналогів, наближаючись до рівномірного шуму.

Порівняльний аналіз отриманих результатів з даними для рекурсивного генератора дозволяє зробити важливий висновок. Властивість квазігрупи бути «експоненційною» (придатною для криптографії) не є безумовною, а розкривається лише за умови використання алгоритму E-перетворень, який забезпечує достатнє розсіювання вхідної ентропії. Успішне відтворення

класифікації Марковського підтверджує валідність розробленого інструментарію тестування та дозволяє використовувати його для подальшого пошуку оптимальних параметрів генерації послідовностей.

2.4.2. Дослідження кореляції між довжиною періоду рекурсивного генератора та властивістю експоненціальності.

В Перевірено гіпотезу про зв'язок між здатністю квазігрупи генерувати максимальний період у простій рекурсивній схемі та її приналежністю до класу «експоненційних». Порівняння вибірки (N=4, T=16) із результатами тестів на основі E-перетворень спростувало цю гіпотезу.

Виявлено два типи розбіжностей:

1. «Лінійні» серед довгоперіодичних: квазігрупи (наприклад, №137, №409) мають ідеальний період, але провалюють статистичні тести через сильну лінійність.
2. Відсутність «експоненційних» у вибірці: якісні структури (№33, №35) у простій схемі часто дають неповні цикли.

Отже, максимальний період у простій рекурсії не є ознакою криптостійкості. Повний цикл може співіснувати з лінійними залежностями, що вимагає застосування складніших архітектур або E-перетворень.

2.5. Аналіз стійкості E-перетворення до низькоентропійних вхідних даних

В ході дослідження було проведено додаткову серію експериментів для перевірки гіпотези про те, що алгебрична структура квазігрупи є домінуючим фактором у формуванні псевдовипадкових властивостей вихідної послідовності, незалежно від якості вхідних даних.

Для цього як вхідний потік («зерно») для E-перетворень використовувалися короткоперіодичні послідовності, згенеровані запропонованим рекурсивним алгоритмом (довжиною L=16 та менше). Ці послідовності, маючи виражену циклічну структуру та низьку ентропію, були

циклічно розширені до необхідного об'єму вибірки та подані на вхід E-перетворення.

Результати моделювання на Торі продемонстрували наступну закономірність:

- У випадку експоненційних квазігруп (наприклад, QG-6, QG-8) вихідна послідовність після E-перетворення (k=50) демонструвала рівномірний розподіл зупинок на Торі («білий шум») та успішно проходила статистичні χ^2-тести. Це свідчить про високу розсіювальну здатність (diffusion property) експоненційних квазігруп, які здатні ефективно знищувати будь-які патерни вхідного повідомлення, перетворюючи їх на псевдовипадковий шум.
- У випадку лінійних квазігруп (наприклад, QG-1, QG-2) вихідна послідовність зберігала та підсилювала структурні закономірності вхідних даних. Візуалізація на Торі демонструвала вироджені траєкторії або чіткі геометричні патерни, що не відповідає вимогам до криптографічних примітивів.

Таким чином, експериментально підтверджено, що експоненційні квазігрупи забезпечують високий рівень нелінійності перетворень, нівелюючи статистичні дефекти джерела вхідних даних.

2.6 Висновки до розділу

У даному розділі було розроблено програмний комплекс та проведено всебічне дослідження властивостей квазігрупових перетворень з метою створення ефективного генератора псевдовипадкових послідовностей (ГПВП). За результатами проведених експериментів зроблено такі висновки:

1. Розроблено метод аналізу простору станів. Створено програмний інструментарій для побудови та аналізу графів станів рекурсивних генераторів. Це дозволило перейти від комбінаторного розгляду латинських квадратів до алгебричного аналізу квазігруп. Виявлено, що топологія графа (кількість

компонентів зв’язності та їх розмір) є первинною характеристикою, що визначає потенціал квазігрупи.

2. Встановлено «Гіпотезу простих чисел». Експериментально доведено, що квазігрупи простих порядків ($N > 3$) мають унікальну алгебричну властивість утворювати єдиний сильно зв’язний граф станів, що дозволяє досягати максимально можливого періоду (N^2+1) для одинарної та N^3+1 для подвійної рекурсії). Натомість квазігрупи складених порядків (N=4, 6, 8) схильні до фрагментації простору станів, що створює ризик зациклення на коротких періодах.

3. Виявлено ефект «колапсу симетрії» у системах N^3. Дослідження рекурсії на двох квазігрупах показало, що використання пари ідентичних квазігруп (L, L) або невдала комбінація (L_1, L_2) призводить до значного погіршення характеристик генератора. Найкращі результати досягаються при використанні структурно різнорідних квазігруп простих порядків.

4. Валідовано метод візуального тестування «Випадкова прогулянка по тору». Проведено верифікацію методу Марковського-Глігороського. Підтверджено ефективність критерію хі-квадрат для розрізнення «лінійних» та «експоненційних» квазігруп при використанні Е-перетворень. Водночас встановлено, що для простих рекурсивних схем цей метод візуалізації виявляє залишкові геометричні патерни навіть для квазігруп вищих порядків, що свідчить про недостатню ентропію простої рекурсії.

5. Спростовано кореляцію між періодом та криптостійкістю: здатність квазігрупи генерувати максимальний цикл у простій рекурсії не гарантує високих статистичних властивостей. Виявлення структур з ідеальним періодом, але лінійною поведінкою, доводить неможливість використання цього параметра як єдиного критерію відбору.

Сформовано вимоги до проєктування вдосконаленого ГПВЧ: використання простих порядків (N=5, 7,…), відмова від простих рекурсивних схем на користь комбінованих підходів або Е-перетворень та обов’язкове поєднання різнорідних квазігруп для уникнення структурних симетрій.

3 СИНТЕЗ АПАРАТНОЇ ОПТИМІЗАЦІЇ ГЕНЕРАТОРА НА ОСНОВІ КВАЗІГРУП

Результати попереднього розділу сформували теоретичний базис для проєктування генераторів псевдовипадкових чисел (ГПВЧ), обґрунтувавши переваги квазігруп простих порядків для забезпечення цілісності графа станів. Водночас, практична реалізація потокових шифрів, особливо в умовах обмежених ресурсів, вимагає використання структур значно більших порядків (зокрема N=256) та високої апаратної ефективності, що унеможливлює пряме застосування простих рекурсивних схем.

У даному розділі вирішується задача масштабування криптографічних примітивів. Розроблено методику побудови квазігруп великих порядків на основі схрещеного добутку та каскадних схем, що дозволяє поєднати високі показники нелінійності з керованою складністю. Окрему увагу приділено апаратній оптимізації генератора шляхом переходу від табличного представлення операцій до системи мінімізованих булевих функцій (поліномів Жегалкіна). Фінальним етапом дослідження є комплексна верифікація статистичних властивостей синтезованих генераторів за допомогою стандартизованого набору тестів NIST SP 800-22, що дозволяє підтвердити їхню відповідність вимогам до криптографічної стійкості.

3.1. Метод побудови квазігруп великих порядків

Для проєктування надійних ГПВЧ критично важливі алгебричні структури великих порядків, що гарантують достатній період гами та ускладнюють криптоаналіз. Оскільки пряма побудова таких квазігруп (зокрема 256×256) обчислювально надскладна, доцільно синтезувати їх із простих компонентів. Перспективним є метод схрещеного добутку, який дозволяє масштабувати простір станів зі збереженням структури. Аби усунути лінійні залежності, притаманні чистому добутку, у роботі застосовано ізотопії. Це підвищує нелінійність, вплив якої досліджено на графі станів та періоді послідовностей

3.1.1. Теоретичні засади методу схрещеного добутку

Метод схрещеного добутку дозволяє побудувати велику квазігрупу з двох менших компонентів: керуючої квазігрупи P і набору локальних квазігруп Q. Конструкція базується на взаємодії двох алгебричних структур різного рівня ієрархії.

Керуюча квазігрупа (P, •) визначає макроструктуру результуючої таблиці операцій, тобто те, як переставляються блоки у великій таблиці. Якщо P має порядок M, то результуюча структура складатиметься з $M \times M$ блоків. Локальні операції представлені множиною квазігруп (Q, Σ), де Σ є набором квазігрупових операцій, кожна з яких визначає мікроструктуру всередині одного блоку. За умови, що Q має порядок N, кожен блок матиме розмір $N \times N$.

Центральним механізмом методу є функція вибору α: $P \times P \rightarrow \Sigma$, яка визначає, яка саме операція з множини Σ застосовується для конкретної пари елементів (p, q) з керуючої квазігрупи. Таким чином, кожна позиція у великій таблиці використовує власну локальну операцію, визначену згідно з відображенням α.

Операція схрещеного добутку для елементів результуючої квазігрупи визначається наступним чином. Елементи мають вигляд (p, a), де $p \in P$ та $a \in Q$, а операція задається рівністю:

$$(p, a) \circledast (q, b) = (p \bullet q, f_{p,q}(a, b)) \qquad (3.1)$$

де $p \bullet q$ є результатом операції в керуючій квазігрупі і визначає номер блоку у результуючій структурі, $f_{p,q}$ позначає локальну операцію, обрану відображенням α для пари (p, q), а $f_{p,q}$ (a, b) є результатом застосування цієї операції до елементів $a, b \in Q$.

Для забезпечення криптографічних властивостей результуюча структура повинна залишатися квазігрупою, що накладає певні обмеження на вихідні компоненти. По-перше, структура (P, •) має бути квазігрупою. По-друге, усі операції з множини Σ мають бути квазігруповими, тобто забезпечувати

оборотність операцій. По-третє, відображення α має бути узгодженим із структурою P та Σ таким чином, щоб гарантувати розв'язність рівнянь у результуючій алгебричнісистемі.

Така конструкція дозволяє отримати квазігрупу порядку $M \times N$ з гнучкою внутрішньою структурою. Варіюючи лише відображення α при незмінних базових компонентах P та Q, можна генерувати значну кількість різних квазігруп, що забезпечує велику варіативність для криптографічних застосувань при збереженні необхідних алгебричних властивостей, таких як відсутність комутативності та асоціативності.

3.1.2. Алгоритмічна реалізація для структур типу 4×2

У рамках даної роботи розроблено програмну реалізацію методу схрещеного добутку, адаптовану для ефективної роботи в сучасних обчислювальних системах. Розглянемо алгоритм побудови квазігрупи порядку 8 на основі базової квазігрупи P порядку 4 та набору операцій Q порядку 2.

Процес формування результуючої матриці 8×8 базується на декомпозиції координат та бітовому керуванні вибором операцій. На першому етапі виконується декомпозиція координат результуючої матриці. Оскільки результуюча структура є композицією блоків, індекси рядка i та стовпця j результуючої матриці, де $i, j \in [0, 7]$, розділяються на дві складові за допомогою операції ділення з остачею. Макро-координати (p_{row}, p_{col}) визначають позицію у керуючій квазігрупі P у діапазоні [0, 3], тоді як мікро-координати (p_{sub_row}, p_{sub_col}) визначають позицію всередині локального блоку Q у діапазоні [0, 1].

На другому етапі здійснюється вибір локальної операції згідно з картою α. Відображення α, яке визначає правило вибору операції для кожної пари макро-координат, зберігається у вигляді цілого числа, що інтерпретується як бітова маска. Така реалізація дозволяє суттєво зменшити обсяг пам'яті порівняно зі зберіганням окремої матриці відповідностей. Для визначення індексу операції обчислюється лінійний індекс блоку:

$$\text{cellidx} = \text{prow} \cdot 4 + \text{pcol} \qquad (3.2)$$

Значення біта у відповідній позиції числа α вказує індекс операції з набору ops_{list}, яку слід застосувати для даної пари макро-координат.

Третій етап полягає в обчисленні компонентів результуючого значення. Значення старшої частини val_p визначається з клітинки квазігрупи P за макро-координатами (p_{row}, p_{col}). Значення молодшої частини val_q обчислюється шляхом застосування обраної на попередньому кроці операції до мікро-координат (p_{sub_row}, p_{sub_col}).

Фінальне значення клітинки результуючої квазігрупи синтезується шляхом лінійної комбінації отриманих компонентів:

$$\text{Result[i][j]=val\_p}\cdot N+\text{val\_q} \qquad (3.3)$$

де N позначає порядок внутрішньої квазігрупи Q. Для розглянутого випадку $N = 2$, що відповідає бітовому зсуву старшої компоненти на один розряд.

Запропонований підхід дозволяє програмно реалізувати швидку генерацію квазігруп великих порядків. Комбінація структур 4×4 та 2×2 забезпечує побудову квазігрупи 8×8, а подальше масштабування до порядків 16×16, 32×32 та вище відбувається за аналогічною логікою. Важливою особливістю методу є те, що експоненційне зростання порядку результуючої структури досягається при лінійному зростанні обчислювальних витрат, що робить алгоритм ефективним для криптографічних застосувань.

3.1.3. Дослідження властивостей синтезованих квазігруп порядку 8

З метою верифікації запропонованого методу схрещеного добутку та визначення оптимальних параметрів побудови криптографічно стійких примітивів, було проведено серію обчислювальних експериментів. Об’єктом дослідження стали квазігрупи порядку $N = 8$, отримані шляхом композиції структур менших порядків (2 та 4).

Основним критерієм оцінки якості синтезованих квазігруп обрано довжину періоду (циклу) псевдовипадкової послідовності.

Враховуючи, що простір станів класичного генератора на одному ЛК для N = 8 становить лише $N^2 = 64$ варіанти, що є недостатнім для статистично значущої оцінки якості перемішування, було вирішено застосувати більш складну схему формування послідовності, де кожен новий елемент залежить не лише від попереднього, а й від кількох попередніх значень. Такий підхід реалізується через рекурентну залежність третього порядку, яка має вигляд:

$$x_i+1=f(x_i,x_i-1,x_i-2) \qquad (3.4)$$

Це дозволяє: наблизити умови тестування до реальної схеми генератора на двох ЛК, описаної у попередніх розділах (що оперує трійками); збільшити простір станів до N^3 що дає змогу виявити приховані лінійні залежності та слабкість блокової структури, які не проявляються на малих періодах. Таким чином, для квазігрупи порядку 8 максимальний розмір простору станів генератора в рамках даного експерименту становить: $8^3 = 512$. Додатково аналізувалися властивості комутативності та симетрії, наявність яких є критичним вразливим місцем для потокових шифрів.

В ході експерименту було розглянуто три сценарії побудови.

Сценарій 1: Декомпозиція 2 × 4 (База порядку 2, локальні операції порядку 4) У цьому експерименті базовою структурою виступали квазігрупи порядку 2, а локальними операціями — квазігрупи порядку 4. Результати моделювання показали високу ефективність даної конфігурації. При переборі доступних карт відображення (α) було отримано квазігрупи з максимальним циклом 252 (що становить майже 50% від теоретичного максимуму простору станів 512). Важливим результатом стало те, що синтезовані таблиці множення не мали властивості симетрії (некомутативні). Це свідчить про те, що схема «мала база — великі локальні операції» сприяє руйнуванню лінійних залежностей у результуючих структурах.

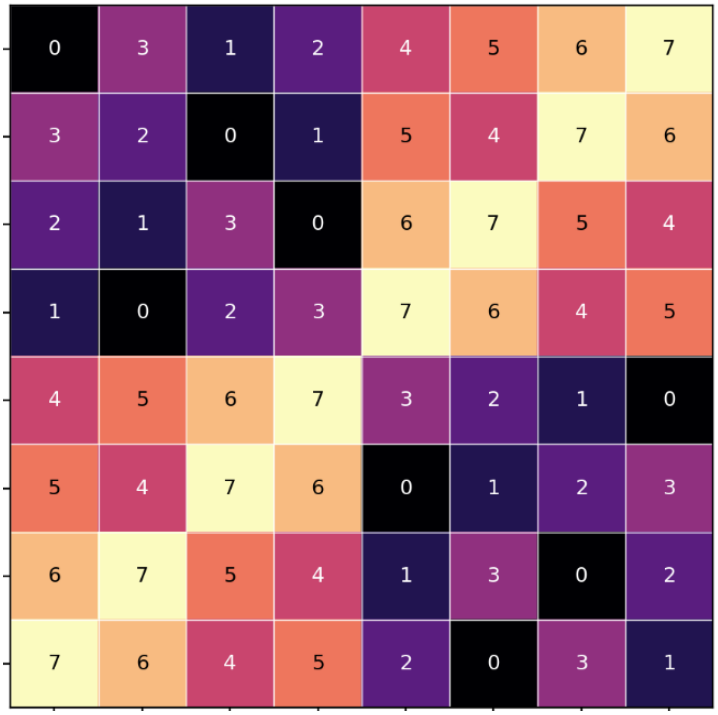


Рис. 3.1 – ЛК8 від 2 × 4 з блоковою симетрією (не комутативний)

Сценарій 2: Декомпозиція 4 × 2 на основі фіксованих груп (Z4 та V4). У другому сценарії роль базової структури виконували класичні групи порядку 4: циклічна група (Z4) та група Клейна (V4). Локальними операціями виступали квазігрупи порядку 2. Аналіз показав незадовільні криптографічні характеристики отриманих структур:

1. Короткі цикли. Максимальна довжина циклу склала лише 16 для бази Z4 та 8 для бази V4.
2. Симетрія. Отримані таблиці виявилися симетричними відносно головної діагоналі, тобто комутативними.

Фізична інтерпретація цього результату полягає у фундаментальних властивостях алгебричних структур. Схрещений добуток, побудований на строгих групах без внесення нелінійності, тяжіє до збереження групових властивостей, зокрема асоціативності. Для ГПВЧ асоціативність є критичним недоліком, оскільки призводить до швидкого замикання траєкторії генератора. Таким чином, використання «чистих» груп як бази є неприпустимим.

Сценарій 3: Декомпозиція 4 × 2 з рандомізацією бази (Ізотопія). Для перевірки гіпотези про вплив перестановок на якість генератора було проведено модифікований експеримент. Базова структура порядку 4 (Z4) була піддана випадковим перестановкам рядків та стовпців (операція ізотопії), що зруйнувало її групову структуру (асоціативність), зберігши при цьому властивість латинського квадрата. Результати підтвердили гіпотезу:

1. Максимальна довжина циклу різко зросла з 16 до 236.
2. Результуючі квазігрупи порядку 8 втратили властивість комутативності (Symmetric: False).

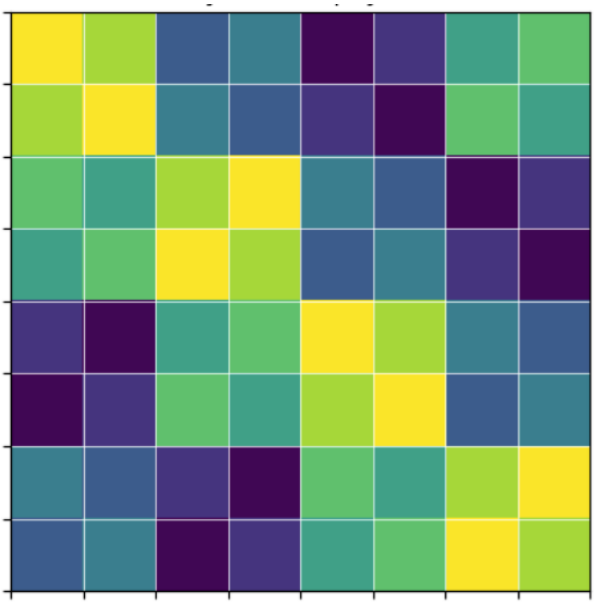


Рис. 3.2 – ЛК8 від 4 × 2 з візуальною симетрією (не комутативний)

Висновки за результатами експерименту проведеного моделювання дозволяє сформулювати ключові вимоги до побудови квазігруп великих порядків методом схрещеного добутку:

1. Метод дозволяє отримувати квазігрупи з довгими періодами (у межах 46-50% від повного простору станів), що є прийнятним показником для проектованих систем.
2. Використання впорядкованих груп (циклічних або абелевих) як базових структур є неефективним через успадкування властивостей асоціативності та комутативності.

3. Критично важливим етапом є попередня рандомізація (шафлінг) компонентів. Застосування випадкових перестановок до базових квазігруп дозволяє уникнути алгебричні виродженості та забезпечити необхідні нелінійні властивості результуючого перетворення.

Отримані результати обґрунтовують доцільність використання схеми 4 × 2 із рандомізованою базою для подальшого масштабування до криптографічних розмірів.

3.1.4. Рекурсивна побудова квазігруп порядку 16 (8×2)

Наступним етапом стала перевірка рекурсивного масштабування квазігруп із порядку 8 до 16 за схемою «8×2». Мета — визначити вплив ізотопії на довжину циклу ГПВЧ та оцінити ефективність методу подвоєння.

Для моделювання використано 16 найкращих квазігруп порядку 8 (база) та квазігрупи порядку 2 (локальні операції). Керування розширенням здійснювала випадкова бінарна матриця 8×8. Оцінку проводили на повному просторі з 4096 станів.

Експеримент включав дві серії. У першій («чистій», 100 ітерацій) базову структуру залишали без змін, варіюючи лише матрицю керування. У другій («ізотопній», 500 ітерацій) до бази щоразу застосовували випадкову ізотопію (перестановку рядків, стовпців та елементів). Результати вимірювання циклів наведено в таблиці 3.1.

Таблиця 3.1 — Порівняльна характеристика довжини циклів для синтезованих квазігруп порядку 16

Номер	Макс. цикл ЛК8	Макс. цикл (Ізотопний ЛК8)	Покриття простору (Ізотоп), %
1	496	2044	~49.9
2	320	2044	~49.9
3	272	2044	~49.9
...	...	...	...
6	56	2036	~49.7
...	...	...	...

Аналіз отриманих даних демонструє суттєву відмінність між двома підходами. Результати чистого сценарію характеризуються високою дисперсією та критично низькою ефективністю. Мінімальні зафіксовані значення циклу для окремих баз склали лише 32 стани, що становить менше 1% від загального простору. Навіть найкращий результат у цій групі (496 станів) забезпечує покриття лише на рівні 12%. Такі показники свідчить про те, що фіксована структура вихідних квазігруп порядку 8 містить внутрішні залежності, які при масштабуванні вступають у конфлікт із картою відображення, призводячи до передчасного замикання циклів генератора.

Застосування ізотопічної рандомізації стабілізує результати в діапазоні 2036–2044 станів. Це підтверджує, що обмеження «чистого» сценарію зумовлені розташуванням елементів, а не природою бази: ізотопія руйнує паразитні кореляції, дозволяючи ефективніше охоплювати простір.

Ключовим результатом є виявлення структурної межі методу 8×2: максимальний цикл склав 2044 при теоретичному максимумі 4096. Це свідчить про сильну імпримітивність та розпад простору на ізольовані орбіти (ймовірно, дві по ~2048), між якими перехід неможливий через блокову архітектуру

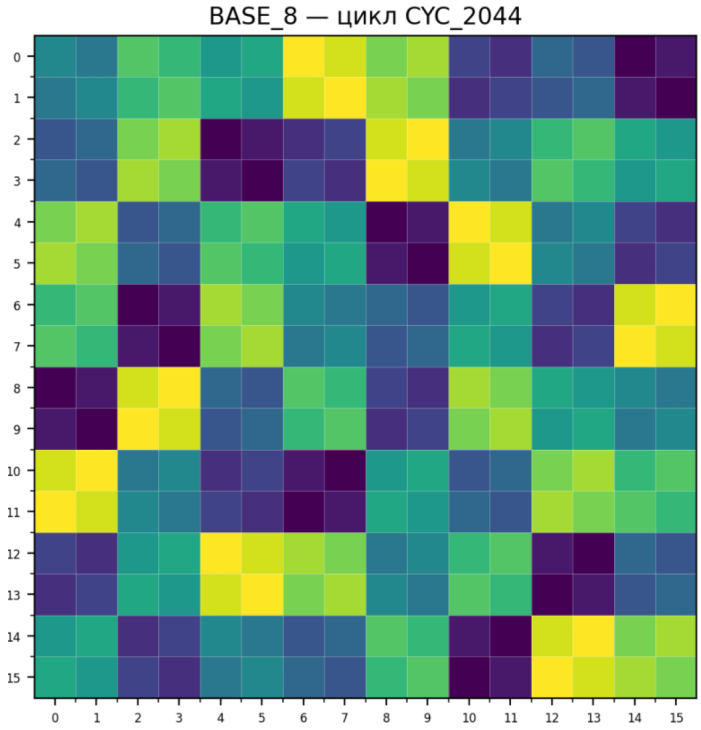


Рис. 3.3 – ЛК16 - візуально не симетричний

Візуальний аналіз таблиць множення отриманих квазігруп підтверджує цей висновок. На хіт-мапі чітко простежується блокова регулярність, відмінна від хаотичного розподілу істинно випадкових латинських квадратів. Хоча отримана квазігрупа не є комутативною, наявність візуальних патернів корелює з обмеженням покриття простору станів рівно наполовину.

Експериментальні результати дозволяють сформулювати кілька важливих висновків щодо практичного застосування методу.

Доведено, що рекурсивна побудова квазігруп є ефективною і не поступається прямим методам, проте просте використання попередніх структур без модифікацій призводить до накопичення структурних залежностей. Критичною умовою для забезпечення стабільних криптографічних властивостей є застосування ізотопних перетворень. Водночас встановлено фундаментальне обмеження методу схрещеного добутку за схемою $N\times2$, який через виникнення ізольованих підпросторів покриває лише ≈50% станів, що вимагає пошуку альтернативних архітектурних рішень для досягнення повного циклу.

3.1.5. Рекурсивна побудова квазігруп порядку 16 (4×4)

Зважаючи на структурні обмеження методу 8×2, виявлені на попередньому етапі (імпримітивність та розділення орбіт, що обмежує цикл на рівні ≈ 50%), було розроблено альтернативну стратегію побудови. Заключний етап експерименту спрямований на синтез квазігруп порядку $N = 16$ шляхом декомпозиції на більш дрібні структурні елементи за схемою 4×4. Гіпотеза дослідження полягала у тому, що збільшення кількості локальних операцій та подрібнення карти відображення дозволить зруйнувати ізольовані підпростори та об’єднати орбіти генератора.

Для реалізації методу 4×4 було визначено наступні параметри конструкції:

- Базова структура (P): Випадкова квазігрупа порядку 4.
- Локальні операції (Q): Набір із 4-х різних випадкових квазігруп порядку 4. Така кількість є оптимальною, оскільки відповідає розмірності бази, дозволяючи використовувати значення карти в діапазоні 0..3.

– Карта відображення (α): Матриця розміром 4×4, елементи якої приймають випадкові значення від 0 до 3.

Загальний простір пошуку карт становить $4^{16} \approx 4{,}3\times10^{9}$ варіантів, що унеможливлює повний перебір. У зв'язку з цим було застосовано метод стохастичного пошуку (Monte Carlo). Алгоритм не фіксує жоден з компонентів: на кожній зі 50 000 ітерацій генерувалася абсолютно нова комбінація бази, набору операцій та карти відображення. Такий підхід дозволяє уникнути локальних оптимумів (поганих районів простору параметрів) та максимізувати ймовірність знаходження ефективної конфігурації.

Динаміка пошуку продемонструвала швидку збіжність до максимальних значень. Основні віхи експерименту та порівняння з попереднім методом наведено в таблиці 3.2.

Таблиця 3.2 — Динаміка росту довжини циклу та порівняння методів побудови

Параметр	Метод 8×2 (попередній)	Метод 4×4 (поточний)
Розмірність карти	8×2 (або 2 × 2)	4×4
Кількість операцій	2	4
Максимальний цикл	2044	3969
Покриття простору	~49.9%	~96.9%
Наявність симетрії	Ні	Ні
Кількість спроб до максимуму	> 1000 (стабільно)	298 (*бистріше*)

Як видно з результатів, уже на 298-й ітерації було досягнуто довжину циклу 3969 станів із теоретично можливих 4096. Подальше продовження експерименту до 150 000 ітерацій не призвело до покращення цього показника, що дозволяє вважати значення 3969 практичною стелею для даної архітектури.

Аналіз показав суттєвий прорив у покритті простору станів. Показник зріс із критичних 49,9% у методі 8×2 до 96,9%, що означає практично повне усунення проблеми розділення орбіт. Втрата близько 3% (127 станів) є прийнятною для криптографічних застосувань, адже ці точки являють собою ізольовані острівці або нерухомі стани, ймовірність потрапляння в які при випадковій ініціалізації мінімальна.

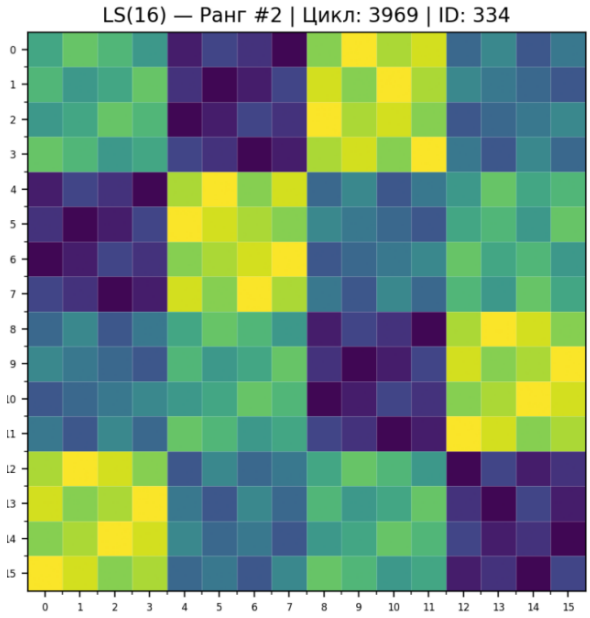


Рис. 3.3 – ЛК16 з 4на4

Успіх досягнуто завдяки зміні топології: висока зернистість методу 4×4 (16 клітинок, 4 операції) дозволила об’єднати малі цикли в єдиний гіперцикл, подолавши обмеження схеми 8×2. Синтезовані квазігрупи є некомутативними та асиметричними, а хаотичний вибір блоків ускладнює зворотний аналіз. Експериментально підтверджено ефективність підходу: досягнуто цикл довжиною 3969, що забезпечує майже повне покриття простору станів. Цю архітектуру рекомендовано для масштабування до N=256.

3.1.6. Рекурсивна побудова квазігруп порядку 64 та 256 ($N\times N$)

Для комплексної перевірки гіпотези про ефективність схеми $N\times N$ при побудові структур великих порядків проведено серію експериментів з синтезу квазігруп порядків 64 та 256 методом схрещеного добутку. Дослідження мало на меті встановити вплив розмірності карти відображення на криптографічні властивості результуючих структур та визначити роль ізотопічних перетворень вихідних компонентів.

Експеримент з побудови квазігруп порядку 64 базувався на схемі 8×8, де базова структура P та всі локальні операції Q мали порядок 8. Карта відображення α реалізовувалася як матриця розміром 8×8 з можливими

значеннями від 0 до 7. Простір станів для оцінки якості генератора становив 64^3 = 262 144 стани. Центральною гіпотезою було припущення, що висока ентропія карти з 64 незалежними клітинками дозволить уникнути проблеми імпримітивності, яка обмежувала метод подвоєння на рівні 50 відсотків покриття.

Експериментальна програма передбачала два контрастні сценарії. У чистому сценарії всі компоненти представляли собою циклічну групу Z_8 у канонічній формі без модифікацій. В ізотопному сценарії використовувалися випадкові ізотопи Z_8, отримані шляхом перестановок рядків, стовпців та перейменування елементів.

Результати чистого сценарію незадовільні: максимальний цикл — 16 станів (0,006% покриття). Це підтверджує збереження алгебричної структури компонентів у схрещеному добутку, де навіть складна карта на 64 клітинки не компенсує слабкість базових елементів.

Ізотопний сценарій продемонстрував протилежні результати. При випадковому пошуку середні значення максимальних циклів становили 150000-200000 станів. Найкращий результат досяг 261121 стан, що відповідає покриттю 99,61 відсотка від теоретичного максимуму. Цей результат було отримано менш ніж за 7000 ітерацій, що свідчить про високу щільність якісних рішень у просторі пошуку при використанні ізотопічних компонентів.

Успішні результати для порядку 64 створили підґрунтя для переходу до порядку 256, який має особливе значення для криптографічних застосувань як стандартний розмір S-box. Конструкція базувалася на схемі 16×16 з використанням квазігруп порядку 16 як вихідних компонентів. Базова структура обиралася серед найкращих квазігруп з попередніх експериментів з подальшим застосуванням ізотопічних перетворень. Відображення α реалізовувалося як матриця 16×16 з 256 клітинками. Простір станів становив 256^3 = 16777216 станів.

Для порядку 256 обчислювальна складність виявилася значно вищою. Початкова стратегія випадкової генерації дозволила досягти циклу 4687200 станів після 31 спроби, що становить 27,94 відсотка покриття. Для підвищення

ефективності запроваджено двоетапну стратегію з використанням простору пар як швидкого критерію фільтрації. Простір пар $N^2 = 65536$ станів є в 256 разів меншим за простір трійок, що дозволяє виконати приблизно 250 попередніх перевірок за час однієї повної. Конфігурації з циклом на просторі пар довшим за 40 000 станів проходили до етапу повної перевірки на просторі трійок.

Перевірено понад 60000 конфігурацій: максимальний цикл сягнув 4682240 станів (27,91% простору). Мутації не забезпечили об'єднання циклів, що свідчить про стійку імпримітивність на великих порядках. Шляхи досягнення покриття понад 90% для порядку 256 викладено в підрозділі 3.1.7, присвяченому каскадним методам.

Порівняльний аналіз результатів для різних порядків (див. табл. 3.3) виявляє нелінійну залежність ефективності методу від розмірності структури. Для порядку 64 досягнуто майже повного покриття на рівні 99,61 відсотка, тоді як для порядку 256 максимальне покриття обмежилося 27,91 відсотка. Це вказує на якісну зміну поведінки системи при переході через критичний порядок у діапазоні між 64 та 256. Для структур порядку 64 зернистість карти 8×8 виявилася достатньою для створення майже повної зв'язності між підпросторами, тоді як для порядку 256 карта 16×16 не забезпечує аналогічного ефекту.

Таблиця 3.3. - Порівняльні характеристики квазігруп різних порядків

Порядок N	Схема	Простір станів N^3	Макс. цикл	Покриття, %	Компоненти
16	8×2	4 096	2 044	49,9	Ізотопи ЛК(8) + ЛК(2)
16	4×4	4 096	3 969	96,9	Ізотопи ЛК(4) + ЛК(4)
64	8×8 (чистий)	262 144	16	0,006	Z_8 без модифікацій
64	8×8 (ізотопи)	262 144	261 121	99,61	Ізотопи Z_8
256	16×16	16 777 216	4 682 240	27,91	Ізотопи ЛК(16)

Результати підтверджують, що метод схрещеного добутку є лише інструментом побудови, тоді як криптографічна стійкість визначається властивостями вихідних компонентів. Застосування ізотопічних перетворень є

необхідною умовою для руйнування алгебричних залежностей. Схема $N \times N$ демонструє значну перевагу над схемою $N \times 2$ для всіх досліджених порядків, забезпечуючи вищу зернистість перемішування та кращу зв'язність простору станів.

3.1.7 Оптимізація каскадних методів для «легких» квазігруп

Результати попереднього підрозділу виявили принципове обмеження методу 16×16 для побудови квазігруп порядку 256, де максимальне покриття простору станів не перевищило 27,91 відсотка навіть при інтенсивному пошуку з використанням еволюційних алгоритмів. Для подолання цього обмеження та одночасного зниження обчислювальної складності конструкції проведено дослідження альтернативних каскадних стратегій побудови з використанням легких квазігруп порядку 4 як базових будівельних блоків.

Вибір квазігруп порядку 4 як фундаментальних елементів конструкції зумовлений результатами повного перебору їх простору, представленими в підрозділі 3.2.2. Серед усіх можливих квазігруп порядку 4 було виявлено кілька унікальних екземплярів (з ідентифікаторами 11, 14, 51 та 54) , які демонструють оптимальне поєднання властивостей:

- Максимальний період генератора: $N^2 - 1 = 15$ станів.
- Мінімальна апаратна складність: Реалізація потребує лише 4 логічних вентилів (згідно з поліномом Жегалкіна).

Використання таких структур як базових блоків дозволяє суттєво знизити апаратні витрати порівняно з використанням випадкових квазігруп або їх ізотопів, які потребують складних таблиць підстановки.

Експериментальна програма передбачала перевірку трьох альтернативних стратегій каскадної побудови квазігруп порядку 256 з різним балансом між криптографічною якістю та обчислювальною складністю. Перша стратегія, що отримала назву послідовного ланцюга або Lightweight Chain, базувалася на рекурсивному застосуванні методу схрещеного добутку за схемою $N \times 4$ без

використання ізотопічних перетворень на проміжних етапах. Послідовність масштабування:

$$4 \rightarrow 16 \rightarrow 64 \rightarrow 256 \qquad (3.5)$$

де на кожному етапі базова структура порядку N комбінувалася з чотирма фіксованими локальними операціями того ж порядку N. Всі компоненти на кожному рівні представляли собою незмінні легкі квазігрупи без застосування перестановок.

Результати першої стратегії виявили феномен структурного насичення, який проявляється у стабілізації відносного покриття простору станів на рівні трохи менше 50 відсотків незалежно від етапу рекурсії (див. Табл. 3.4).

- Для порядку 16: 49.22% (2016 з 4096).
- Для порядку 64: 48.45% (127 008 з 262 144).
- Для порядку 256: 47.69% (8 001 504 з 16 777 216).

Ефект стабілізації покриття свідчить, що метод $N\times4$ без ізотопій діє аналогічно $N\times2$, обмежуючи простір станів рівнем ~50%. Ця закономірність, підтверджена на трьох рівнях масштабування, є внутрішньою властивістю конструкції. Відсутність ізотопічних перетворень зберігає алгебричні залежності, що перешкоджає повній зв'язності простору незалежно від глибини рекурсії.

Стабільність цього ефекту на трьох рівнях масштабування свідчить про внутрішню властивість конструкції, а не випадкову флуктуацію. Відсутність структурного хаосу через ізотопічні перетворення зберігає алгебричні залежності, що обмежують зв’язність простору.

Практичне значення виявленої закономірності дозволяє створювати апаратно ефективні криптографічні примітиви з гарантованими, хоча й не оптимальними властивостями. Для структур порядку 256 забезпечується стабільний цикл ≈8 млн станів, використовуючи лише один модуль 4×4 та набір карт відображення. Це компроміс між високим покриттям (>99%) і апаратною

вартістю, прийнятний для застосувань, де важливий баланс між стійкістю та ресурсами.

Таблиця 3.4. Порівняльні характеристики каскадних стратегій побудови ЛК-256

Стратегія	Послідовність	Ізотопія	Макс. цикл	Покриття, %	Складність синтезу	Апаратна вартість
Lightweight Chain	4→16→64→256	Ні	>8 млн	47,69	Низька	Мінімальна
Isotopic Chain	4→16→64→256	На кожному рівні	>16 млн	>98	Висока	Висока
Heavyweight 16×16	4→16, потім 16×16→256	На етапі 4→16	>15 млн	>90	Середня	Середня

Стратегія «ізотопного ланцюга» (рекурсія 4→256 із випадковими ізотопіями на кожному рівні) руйнує алгебраїчні залежності, забезпечуючи цикл понад 16,4 млн станів (>98% покриття). Головні недоліки — висока обчислювальна складність та значна апаратна ємність через необхідність зберігання таблиць підстановок.

Композитний метод (Heavyweight 16×16) передбачає попередній відбір квазігруп порядку 16 (із покриттям >95%) та їх використання як бази й локальних операцій для синтезу порядку 256. Це забезпечило баланс якості (цикл >15,2 млн станів, покриття >90%) і швидкодії, оскільки ресурсномісткий пошук виконується на просторі меншої розмірності, а фінальне формування структури відбувається миттєво.

У результаті вибір методики має визначатися цільовим балансом між апаратною вартістю та покриттям простору станів: від мінімального кремнію з прийнятним циклом у Lightweight Chain до максимальної зв’язності в Isotopic Chain і збалансованої продуктивності в Heavyweight 16×16.

3.2. Апаратний синтез логічної структури генератора

Процес імплементації потокового шифру на базі квазігруп вимагає трансформації абстрактного математичного опису операцій у фізичну архітектуру кристала. Пряме використання таблиць множення (таблиць Келі) для квазігруп великих порядків є нераціональним з точки зору використання ресурсів

кристала, оскільки споживання пам'яті зростає квадратично відносно порядку квазігрупи. У контексті проектування спеціалізованих інтегральних схем (ASIC) або програмованих логічних матриць (FPGA), ключовим завданням стає мінімізація площі кристала та енергоспоживання при збереженні високої криптографічної стійкості [20]. Тому перехід від табличного опису (Look-Up Table, LUT) до логічних функцій є необхідним етапом синтезу, що дозволяє замінити блоки пам'яті (BRAM/ROM) на комбінаційні мережі логічних вентилів.

Розробка апаратної структури генератора базується на принципах побудови надійних цифрових автоматів, досліджених у працях В. А. Лужецького [25]. Зокрема, використання мінімізованих логічних схем (поліномів Жегалкіна) дозволяє підвищити відмовостійкість системи порівняно з табличними методами (S-box).

3.2.1. Перехід від таблиць до поліномів Жегалкіна (ANF).

Будь-яка бінарна операція квазігрупи порядку N, де N є степенем двійки (2^k), може бути представлена як система з k булевих функцій від $2k$ змінних. У контексті апаратної реалізації на FPGA, таблиця істинності такої системи природним чином відображається у конфігуровані логічні блоки (LUT), які фактично є невеликими модулями пам'яті. Проте альтернативним і часто більш ефективним підходом є представлення цих функцій у вигляді алгебричної нормальної форми (ANF), відомої як поліноми Жегалкіна. Таке представлення дозволяє реалізувати операцію множення квазігрупи через мережу логічних елементів XOR (додавання за модулем 2) та AND (логічне множення) [21].

Для оцінки ефективності апаратної реалізації вводиться метрика апаратної вартості (Cost). У технології ASIC ця величина вимірюється в еквівалентних вентилях (Gate Equivalents, GE), де за одиницю зазвичай приймається площа двоходового елемента NAND. У технології FPGA метрикою слугує кількість задіяних LUT або слайсів. Існує пряма залежність між складністю полінома Жегалкіна — його алгебрчним степенем та кількістю мономів — і апаратною вартістю реалізації. Мінімізація кількості термів у поліномі дозволяє знизити

затримку розповсюдження сигналу та динамічне енергоспоживання пристрою [22].

Гіпотеза етапу пов'язує логічну складність квазігрупи з її криптографічними властивостями. Передбачається, що прості поліноміальні форми зумовлюють короткі цикли та слабку дифузію, тоді як висока алгебрична складність покращує статистичні характеристики, але збільшує вимоги до апаратних ресурсів. Встановлення балансу між цими параметрами є критичним для створення ефективних легких криптографічних примітивів [23]

3.2.2. Експериментальне дослідження простору порядку 4.

Для верифікації гіпотези про взаємозв'язок між апаратною складністю та криптографічними властивостями проведено повний перебір простору квазігруп порядку 4. Оскільки квазігрупа порядку 4 оперує з двобітовими значеннями, її операція описується двома булевими функціями від чотирьох змінних, що дозволяє exhausive аналіз всіх можливих структур з подальшою мінімізацією їх поліноміальних форм Жегалкіна.

В ході повного перебору виявлено групу квазігруп, зокрема з ідентифікаторами 76 та 99, які реалізуються рекордно низькою кількістю ресурсів у 3 логічні вентилі XOR і при цьому забезпечують максимальний період генератора $N^2 - 1 = 15$. Аналіз поліноміальних форм цих структур виявив їх лінійну природу. Для квазігрупи ID 76 логічні функції мають вигляд:

$$\begin{cases} z_0 = k_1 \oplus m_0 \\ z_1 = k_0 \oplus k_1 \oplus m_1 \end{cases} \quad (3.5)$$

де k_0, k_1 представляють біти першого операнда, m_0, m_1 — біти другого операнда, *а* z_0, z_1 — біти результату. Відсутність термів з операцією кон'юнкції свідчить про алгебричний ступінь, що дорівнює 1.

Лінійність таких структур робить їх вразливими до алгебричних атак. Генератор на основі лінійних булевих функцій може бути змодельований системою лінійних рівнянь над GF(2), стан якої відновлюється за декілька спостережень вихідної послідовності з використанням алгоритму Берлекемпа-

Мессі. Це унеможливлює їх використання для криптографічних застосувань незважаючи на мінімальну апаратну вартість та максимальний період.

Для забезпечення криптографічної стійкості необхідна нелінійність булевих функцій, що досягається наявністю термів з операцією кон'юнкції. Найближчим кандидатом, що поєднує максимальний період 15 та нелінійність, виявилася квазігрупа з ідентифікатором 11. Її апаратна вартість становить 4 логічні вентилі, що лише на один більше за лінійні аналоги. Поліноміальне представлення має вигляд:

$$\begin{cases} z_0 = k_0 \oplus m_1 \\ z_1 = k_1 \oplus m_0 \oplus (k_0 \cdot m_1) \end{cases} \quad (3.6)$$

Наявність терму $k_0 \cdot m_1$ забезпечує квадратичний алгебричний ступінь другої функції, що робить систему нелінійною та стійкою до алгебричних атак лінійної складності.

Результати повного перебору візуалізовано на рисунку 3.5, який демонструє залежність періоду генератора від апаратної вартості реалізації. На графіку виділяються три характерні області з різною поведінкою системи.

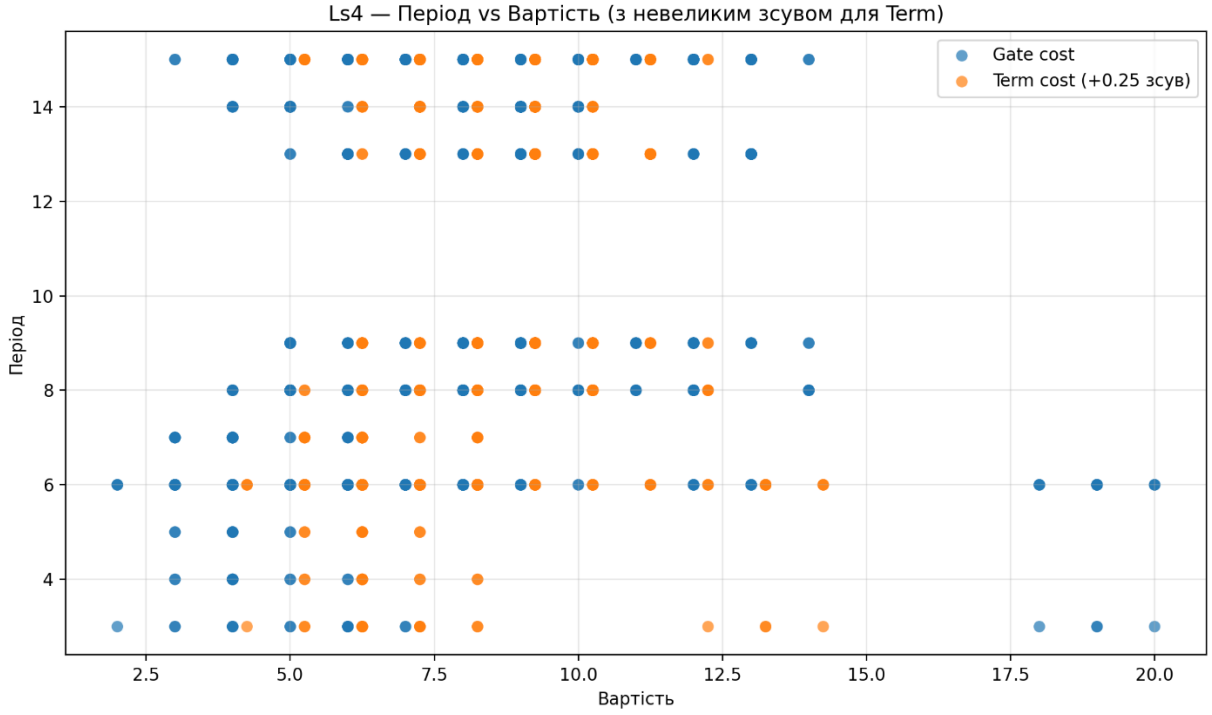


Рис. 3.5. Залежність періоду генератора від апаратної складності.

Виділено три характерні області розподілу. Перша (Gate ≈ 4–5, Term ≈ 5–6) демонструє оптимальний баланс ресурсів: максимальний період 15 досягається вже при мінімальній складності (Gate = 4, Term = 5), прикладом чого є квазігрупа ID 11. Друга область — щільний середній кластер (Gate ≈ 8–12, Term ≈ 8–10) із розкидом періодів 6–15; тут зростання вартості до рівня (10,10) підвищує медіану періоду, проте подальше ускладнення не гарантує приросту ефективності. Третя область (Gate ≥ 18) містить «патологічні» конфігурації, де попри високу вартість генеруються короткі цикли (3–6). Статистичний аналіз підтвердив відсутність значущого зв'язку між апаратною вартістю та періодом (кореляції Пірсона -0,006, Спірмена 0,097), що вказує на співіснування структур із полярно різною ефективністю в межах однакової складності.

Таблиця 3.6 – Порівняння ЛК

ID квазігрупи	Апаратна вартість (вентилів)	Період циклу (T)	Алгебрична властивість	Рішення
76	3	15	Лінійна (deg=1)	Відхилено (Криптографічно нестійка)
11	4	15	Квадратична (deg=2)	Обрано (Оптимальне співвідношення)
12	6	15	Квадратична (deg=2)	Відхилено (Апаратно неефективна)
Випадкова	~10–12	< 15	Змінна	Відхилено (Низька якість)

Порівняльний аналіз характерних представників (табл. 3.6) виявляє компромісні співвідношення між різними характеристиками:

- Лінійна квазігрупа ID 76 з вартістю 3 гейти досягає максимального періоду 15, але її алгебричний ступінь 1 робить її вразливою до алгебричних атак та відновлення стану методом Берлекемпа-Мессі
- Квазігрупи по типу ID 11, що має апаратну вартість лише 4 гейти, поєднує максимальний період 15 з квадратичним алгебричним ступенем, забезпечуючи достатню нелінійність при мінімальній складності. Її використання дозволяє реалізувати принцип псевдо-недетермінованого захисту [24], коли криптографічна стійкість ґрунтується не лише на

секретному ключі, а й на унікальних параметрах апаратної реалізації конкретного екземпляра шифру

- Квазігрупа ID 12 з вартістю 6 гейтів також має квадратичний ступінь та період 15, Проте є апаратно неефективною порівняно з 11
- Випадково обрані квазігрупи демонструють середню складність 10-12 гейтів з непередбачуваними періодами, часто меншими за максимум

Структура типу ID 11 є оптимальною для N=4, поєднуючи максимальний період із мінімальною складністю. Попри обмеження квадратичного ступеня, вона придатна як базовий елемент, оскільки подальше каскадування підвищує загальну нелінійність. Прямий перебір для порядків 5, 6, 8 визнано недоцільним через експоненційне зростання варіантів та неефективність відображення недвійкових розмірностей в FPGA. Тому масштабування системи до вищих порядків варто виконувати методом композиції на основі знайденого примітиву..

3.3. Статистичний аналіз та верифікація за стандартами NIST

Для підтвердження криптографічної якості розроблених методів побудови квазігруп та алгоритмів генерації псевдовипадкових послідовностей проведено серію статистичних тестів згідно зі стандартом NIST SP 800-22 rev 1a. Тестування виконувалося на вибірках обсягом 10^6 біт з використанням програмної реалізації тестового пакета, доступної у відкритому репозиторії github.com/stevenang/randomness_testsuite

Експериментальна програма охоплювала порівняння різних стратегій побудови квазігруп та архітектур генераторів для визначення оптимальної конфігурації системи. У дослідженні розглядалися три основні стратегії побудови квазігруп порядку 256: легкий ланцюг (Lightweight Chain), ізотопний ланцюг (Isotopic Chain) та композитний метод (Heavyweight 16×16). Кожна стратегія тестувалася у двох архітектурах генераторів, що відрізняються глибиною композиції операцій.

- (Gen2) Рекурсія на одному ЛК $u_i = L[L[u_{i-1}][u_{i-2}]][u_{i-3}]$
- (Gen3) Композиція двох ЛК $u_i = L_2[L_1[u_{i-1}][u_{i-2}]][u_{i-3}]$

Додатково досліджувалися генератори на базі квазігруп меншого порядку 64 для визначення мінімально допустимої розмірності структур.

3.3.1. Вплив архітектури генератора на статистичні властивості

Перший етап експериментального дослідження мав на меті встановити вплив кількості квазігруп у композиції на статистичні властивості вихідної послідовності. Центральною гіпотезою було припущення про те, що використання однієї квазігрупи, навіть високої якості, може бути недостатнім для забезпечення необхідного рівня ентропії через збереження структурних залежностей на рівні окремих операцій.

Для композитного методу побудови квазігрупи з використанням елітних блоків 16×16 отримано наступні результати. Архітектура Gen2 з однією квазігрупою продемонструвала проходження базових тестів, зокрема Frequency (Monobit) з P-value = 0,983 та Linear Complexity з P-value = 0,933. Водночас спостерігалися критичні провали у тестах Overlapping Template (P-value = 0,004), Approximate Entropy (P-value = 0,009) та Discrete Fourier Transform (P-value ≈ 0). Архітектура Gen3 з композицією двох квазігруп усунула більшість виявлених проблем: тести Overlapping Template (P-value = 0,754) та Approximate Entropy (P-value = 0,267) успішно пройдено, тоді як тест DFT залишився непройденим незалежно від архітектури.

Аналогічна закономірність спостерігалася для стратегії ізотопного ланцюга. Архітектура Gen2 систематично провалювала тести Overlapping Template (P-value = 0,0003), Approximate Entropy (P-value = 0,002) та Serial Test (P-value ≈ 0). Перехід до архітектури Gen3 забезпечив успішне проходження цих тестів з високими показниками: Overlapping Template (P-value = 0,355), Approximate Entropy (P-value = 0,971) та Serial Test (P-value = 0,232).

Виявлені закономірності вказують на системну недостатність архітектури Gen2 незалежно від методу побудови базової квазігрупи. Провали тестів на ентропію та пошук шаблонів свідчать про наявність прихованої мікроперіодичності та недостатнє перемішування на рівні трійок послідовних

станів. Використання композиції двох різних квазігруп у архітектурі Gen3 забезпечує додатковий рівень дифузії, що руйнує залишкові алгебричні залежності та підвищує статистичну якість вихідної послідовності.

3.3.2. Порівняльний аналіз методів побудови квазігруп

Другий етап дослідження було присвячено порівнянню двох стратегій побудови квазігруп порядку 256 в архітектурі Gen3.

- Метод блоків 16×16 показав стабільні результати, проте значення P-value для тесту ентропії (~0.26) свідчить лише про середню якість розподілу.
- Метод ізотопного ланцюга продемонстрував значно кращі показники: Frequency ~0.65, Runs ~0.71, Linear Complexity ~0.64, Approximate Entropy ~0.97. Це вказує на майже ідеальну статистичну структуру та високу дифузію бітів.

Ізотопний ланцюг виявився ефективнішою стратегією, оскільки дрібна зернистість перемішування на рівнях 4→16→64→256 забезпечує якісніший хаос у порівнянні з блоковим підходом.

3.3.3. Верифікація стабільності результатів

Для підтвердження надійності та незалежності результатів від конкретного вибору квазігруп проведено серію перехресних тестів різних комбінацій структур. Відібрано три найякісніші квазігрупи порядку 256, побудовані методом ізотопного ланцюга, позначені як LC1, LC2 та LC3. Протестовано шість можливих пар композицій у архітектурі Gen3, включаючи прямі та зворотні комбінації.

Результати перехресного тестування представлено в таблиці 3.4. Усі шість комбінацій успішно пройшли 14 з 15 базових тестів NIST. Показники для ключових тестів демонструють високу стабільність: Frequency в діапазоні 0,180-0,930, Runs в діапазоні 0,360-0,990, Linear Complexity в діапазоні 0,220-0,990, Approximate Entropy в діапазоні 0,390-0,980. Жодна комбінація не показала

провалів у тестах на ентропію чи лінійну складність, що підтверджує надійність методу.

Таблиця 3.4. Результати перехресного тестування композицій квазігруп

Комбінація	Frequency	Runs	Linear Complexity	Approximate Entropy	Spectral	Загальний статус
LC1 + LC2	0,760	0,920	0,220	0,690	0,000	14/15 PASS
LC1 + LC3	0,280	0,960	0,990	0,640	0,000	14/15 PASS
LC2 + LC1	0,720	0,640	0,910	0,980	0,000	14/15 PASS
LC2 + LC3	0,180	0,980	0,660	0,710	0,000	14/15 PASS
LC3 + LC1	0,930	0,360	0,530	0,690	0,000	14/15 PASS
LC3 + LC2	0,460	0,990	0,810	0,390	0,000	14/15 PASS

Важливим спостереженням є відмінність результатів для прямих та зворотних пар. Комбінації (LC1, LC2) та (LC2, LC1) дають статистично різні, хоча й Протеово високі результати, що підтверджує некомутативність композиції квазігрупових операцій. Ця властивість має практичне значення для збільшення простору ключів: кожна перестановка порядку квазігруп у композиції генерує статистично незалежну псевдовипадкову послідовність.

Єдиним тестом, який стабільно не проходить для всіх комбінацій, є Discrete Fourier Transform з P-value ≈ 0. Систематичність цього результату вказує на фундаментальну властивість алгебричних генераторів на базі квазігруп. Провал спектрального тесту свідчить про наявність певних частотних компонент у вихідній послідовності, що може бути пов'язано з алгебричною структурою квазігрупових операцій. Для більшості криптографічних застосувань, де критичними є тести на ентропію та лінійну складність, цей недолік не є критичним, оскільки компенсується відмінними показниками в інших категоріях.

3.3.4. Аналіз структур меншого порядку та ранніх підходів

Для оцінки меж застосовності методів досліджено генератори на базі квазігруп порядку 64 та ранні варіанти структур порядку 256 без оптимізацій.

Експеримент з квазігрупами порядку 64 (максимальний цикл ~$2{,}6\times10^5$ станів) показав критичне обмеження: для вибірки 10^6 біт послідовність довелося повторювати чотири рази. Архітектура Gen2 з однією квазігрупою порядку 64 провалила тести Runs (0,0009), Longest Run (≈0), DFT (≈0), Overlapping Template (0,0006) та Approximate Entropy (≈0), що підтверджує недостатність такої довжини циклу. Gen3 з двома квазігрупами порядку 64 покращила результати (Frequency = 0,719, Linear Complexity = 0,923), але провалила Runs, DFT та Approximate Entropy, тому лишається непридатною для криптографії порівняно з системами порядку 256 (де Approximate Entropy = 0,970).

Ранні експерименти з квазігрупами порядку 256, побудованими методом схрещеного добутку 16×16 без оптимізації, також виявили обмеження. Gen2 провалила Overlapping Template (0,003), Approximate Entropy (0,005) та DFT (≈0), що підтверджує слабкість однокомпонентної архітектури. Gen3 пройшла тести на ентропію (0,620) та лінійну складність (0,170), але покриття простору станів склало лише 28%, тоді як метод ізотопного ланцюга забезпечує понад 98%.

3.3.5. Узагальнення результатів та визначення оптимальної конфігурації

На основі аналізу п'ятнадцяти наборів результатів тестування різних конфігурацій генераторів встановлено оптимальну комбінацію параметрів системи. Найвищі показники за сукупністю критеріїв довжини циклу, ентропії та лінійної складності демонструє конфігурація, що поєднує метод ізотопного ланцюга для побудови квазігруп порядку 256 та архітектуру Gen3 з композицією двох різних квазігруп. Усереднені показники P-value для цієї конфігурації за групою успішних тестів становлять: Frequency = 0,650, Block Frequency = 0,580, Runs = 0,710, Longest Run = 0,620, Linear Complexity = 0,640, Approximate Entropy = 0,970, Overlapping Template = 0,350, Serial Test = 0,590. Беззаперечним лідером за сукупністю показників (довжина циклу, ентропія, лінійна складність) визнано комбінацію:

– Метод побудови ЛК: ізотопного ланцюга (4 → 16 → 64 → 256 з повною ізотопією).

– Архітектура генератора: Тип 3 (Gen3) — композиція двох квазігруп

Таблиця 3.5. Порівняння усереднених показників P-value для основних конфігурацій

Тест NIST	Gen2 (один ЛК)	Gen3 (композитний)	Gen3 (ізотопний ланцюг)
Frequency (Monobit)	0,980	0,530	0,650
Block Frequency	0,420	0,610	0,580
Runs Test	0,350	0,450	0,710
Longest Run	0,550	0,680	0,620
Linear Complexity	0,930	0,190	0,640
Approximate Entropy	<0,010	0,260	0,970
Overlapping Template	<0,010	0,750	0,350
Serial Test	<0,010	0,230	0,590
Spectral (DFT)	<0,010	<0,010	<0,010
Загальна оцінка	Незадовільно	Задовільно	Відмінно

Статистичні тести підтвердили ефективність методів. Для криптографії рекомендовано квазігрупи порядку 256+, побудовані ізотопним ланцюгом із повним перемішуванням. Для належної ентропії архітектура має поєднувати мінімум дві різні квазігрупи. Провал спектрального тесту вказує на потребу в дослідженні частотних характеристик, проте не є критичним для більшості практичних задач.

3.3.6. Порівняльний аналіз із сучасними аналогами

Для оцінки конкурентоспроможності розробленого генератора проведено порівняльний аналіз його характеристик з відомими криптографічними генераторами псевдовипадкових чисел. До порівняння включено стандартизовані алгоритми: ChaCha20 (потоковий шифр на основі ARX-конструкцій, стандарт TLS 1.3), AES-CTR (блоковий шифр у режимі лічильника), HC-128 (фіналіст eSTREAM) та Trivium (переможець eSTREAM у категорії апаратної реалізації).

Порівняння проводилося за трьома групами критеріїв: статистична якість вихідної послідовності за тестами NIST SP 800-22, обчислювальна ефективність

та криптографічна стійкість. Результати статистичного тестування представлено в таблиці 3.7.

Таблиця 3.7. Порівняння проходження тестів NIST для різних генераторів

Генератор	Пройдено тестів	Провалено тестів	Критичні провали	Approximate Entropy	Linear Complexity
Розроблений (Gen3, Isotopic Chain)	14/15	1/15	DFT	0,970	0,640
ChaCha20	15/15	0/15	—	>0,99	>0,99
AES-CTR	15/15	0/15	—	>0,99	>0,99
HC-128	15/15	0/15	—	>0,99	>0,99
Trivium	15/15	0/15	—	>0,99	>0,99

Розроблений генератор демонструє виключно високі показники для тестів на ентропію (P-value = 0,970) та лінійну складність (P-value = 0,640), що практично відповідає рівню стандартизованих аналогів. Єдиним непройденим тестом є Discrete Fourier Transform, що вказує на наявність спектральних особливостей, притаманних алгебричним генераторам на основі квазігруп.

Внутрішній стан розробленого генератора визначається трьома послідовними байтами та двома квазігрупами порядку 256, що становить приблизно 1 мегабіт, значно перевищуючи розмір стану ChaCha20 (512 біт), AES-CTR (256 біт), HC-128 (4096 біт) та Trivium (288 біт). Великий розмір стану потенційно підвищує стійкість до атак на основі аналізу послідовностей, Проте створює додаткові вимоги до пам'яті.

Обчислювальна ефективність розробленого генератора поступається стандартизованим аналогам через необхідність трьох табличних підстановок розміром 256×256 байт, що вимагає 192 кілобайт пам'яті. Для порівняння, ChaCha20 використовує лише арифметичні операції, AES-CTR потребує 160 байт для S-box, а Trivium оперує виключно бітовими операціями. На сучасних процорах швидкість ChaCha20 перевищує 1000 МБ/с, тоді як табличні підстановки великого розміру обмежують швидкість розробленого генератора через неефективне використання кешу процесора.

Ключові переваги генератора — алгебрична модульність та гнучкість каскадної архітектури (4→256). Оптимальна конфігурація забезпечує цикл понад 16 млн станів. Завдяки базовим блокам порядку 4 (лише 4 логічні елементи) метод є ефективним для реалізації на FPGA та у вбудованих системах.

Основне обмеження — відсутність тривалої історії криптоаналізу, властивої стандартам (ChaCha20, AES, Trivium), що частково компенсується можливістю формальної верифікації. Поступаючись ARX-алгоритмам у програмній швидкодії, розроблений генератор не програє у статистичних показниках. Його рекомендовано для спеціалізованих застосувань із пріоритетом на апаратну оптимізацію та доказову стійкість.

3.4. Висновки до розділу

У даному розділі розроблено та експериментально верифіковано методи синтезу квазігруп великих порядків для побудови криптографічно стійких генераторів псевдовипадкових послідовностей. Основні результати дослідження:

1. Розроблено метод каскадної побудови квазігруп великих порядків. Запропоновано схему рекурсивного схрещеного добутку 4 → 16 → 64 → 256 з використанням легких квазігруп порядку 4 як базових будівельних блоків. Експериментально встановлено, що схема $N \times N$ забезпечує значно вищу ефективність (покриття простору станів 96,9% для N=16, 99,6% для N=64) порівняно зі схемою $N \times 2$ (обмеження на рівні 50%).

2. Виявлено ефект структурного насичення для методу подвоєння. Доведено, що каскадна побудова без ізотопічних перетворень стабілізує покриття простору станів на рівні ~48% незалежно від кількості рівнів масштабування. Це підтверджує критичну роль ізотопії для руйнування алгебричних залежностей між рівнями ієрархії.

3. Запропоновано три стратегії побудови квазігруп порядку 256 з різним балансом між криптографічною якістю та апаратною вартістю:

- Lightweight Chain (послідовний ланцюг): забезпечує покриття 47,7% при мінімальній апаратній складності, придатний для ресурсообмежених систем.
- Isotopic Chain (ізотопний ланцюг): досягає покриття >98% через застосування перетворень на кожному рівні, оптимальний для максимальної криптостійкості.
- Heavyweight 16×16 (композитний метод): забезпечує покриття >90% при помірній складності, рекомендований для практичних застосувань.

4. Здійснено апаратну оптимізацію базових компонентів. Методом повного перебору простору квазігруп порядку 4 ідентифіковано оптимальну структуру (ID 11) з квадратичним алгебричним степенем, максимальним періодом ($N^2-1=15$) та мінімальною апаратною вартістю (4 логічні вентилі). Встановлено відсутність прямої кореляції між апаратною складністю та довжиною періоду (коефіцієнт Пірсона $\approx -0{,}006$), що обґрунтовує необхідність цілеспрямованого відбору компонентів.

5. За стандартом NIST SP 800-22 оптимальна конфігурація (Isotopic Chain + Gen3) успішно пройшла 14 з 15 тестів із високими показниками. Єдиний непройдений тест (DFT) є системною особливістю алгебричних генераторів, що компенсується іншими характеристиками.

6. Доведено критичну роль архітектури Gen2 (рекурсія однієї квазігрупи) систематично провалює тести на ентропію, тоді як Gen3 (композиція двох різних квазігруп) забезпечує необхідну дифузію та усуває структурні залежності.

7. Підтверджено некомутативність композиції, порядок застосування квазігруп впливає на властивості, розширюючи простір ключів. Розроблені методи дозволяють балансувати між стійкістю та ресурсами: для критичних задач рекомендовано Isotopic Chain + Gen3 (>98% покриття), для ресурсообмежених — Lightweight Chain (~48%).

4 ЕКОНОМІЧНА ЧАТИНА

Під час проведення наукових досліджень необхідно враховувати як потенційні витрати на проведення дослідницького процесу, так і безпосередні результати, які визначають доцільність проведення дослідження. Отримані результати характеризують створений кінцевий продукт, а також наукові знання, які можуть бути застосовані для подальшого розвитку науки і техніки.

Комплексна магістерська кваліфікаційна робота на тему “Метод та засіб потокового шифрування на основі квазігруп”, а саме її друга частина “Операційний блок”, відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

Для розрахунку доцільності та ефективності інвестицій, вкладених в дану роботу, необхідно провести такі етапи робіт:

1. проведення комерційного та технологічного аудиту науково-технічної розробки, тобто встановлення її науково-технічного рівня та комерційного потенціалу;
2. розрахунок витрат на здійснення науково-технічної розробки;
3. розрахунок економічної ефективності науково-технічної розробки у випадку її можливої комерціалізації потенційним інвестором та обґрунтування економічної доцільності такої комерціалізації.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення аудиту є оцінювання науково-технічного рівня та комерційного потенціалу розробленого методу та засобів потокового шифрування. Оцінювання здійснюється за 12 критеріями за 5-бальною шкалою експертним шляхом [26]. Критерії для оцінки науково-технічного рівня і комерційного потенціалу розробки наведені у таблиці 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
Кри-терій	1	2	3	4	5
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних умовах
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає

Продовження таблиці 4.1

Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь- які регламентні обмеження на виробництво та реалізацію продукту

Для проведення оцінювання необхідно визначити оцінки відповідно кожного критерія і обчислити середню арифметичну оцінку, яка визначатиме науково-технічний рівень та комерційний потенціал розробки.

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки наведено у таблиці 4.2.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	5	5	5
2. Ринкові переваги (наявність аналогів)	5	4	4
3. Ринкові переваги (ціна продукту)	5	5	5
4. Ринкові переваги (технічні властивості)	4	4	4
5. Ринкові переваги (експлуатаційні витрати)	5	5	5
6. Ринкові перспективи (розмір ринку)	3	3	3
7. Ринкові перспективи (конкуренція)	3	3	3
8. Практична здійсненність (наявність фахівців)	4	4	4
9. Практична здійсненність (наявність фінансів)	2	3	2
10. Практична здійсненність (необхідність нових матеріалів)	4	4	4
11. Практична здійсненність (термін реалізації)	4	4	4
12. Практична здійсненність (розробка документів)	3	3	3
Сума балів:	47	47	46
Середньоарифметична сума балів $СБ_с$	46.67		

Середньоарифметична сума балів оцінюється за наступною формулою:

$$СБ_с = \frac{\sum_{i=1}^{3} СБ_i}{3} = \frac{47 + 47 + 46}{3} = 46.67 \qquad (4.1)$$

де $СБ_с$ – середньоарифметична сума балів; $СБ_i$ – сума балів і-го експерта.

За результатами розрахунків, наведених в таблиці 4.2, можна зробити висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки [26]. Висновок ґрунтується на оцінках кожного рівня, що наведені в таблиці 4.3

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41…48	Високий
31…40	Вищий середнього
21…30	Середній
11…20	Нижчий середнього
0…10	Низький

В Отриманий результат 46.67 бала свідчить про те, що науково-технічний рівень та комерційний потенціал розробки є високим. Такий рівень забезпечується передусім інноваційністю запропонованого методу синтезу квазігруп (Isotopic Chain) та оптимізованою архітектурою генератора (Gen3). Теоретично обґрунтована та програмно підтверджена можливість досягнення високих криптографічних показників (успішне проходження тестів NIST SP 800-22) при мінімальній алгоритмічній складності створює значні конкурентні переваги продукту.

Високі оцінки за критеріями «Технічна здійсненність» та «Експлуатаційні витрати» зумовлені тим, що розроблений метод дозволяє реалізувати надійне шифрування на ресурсообмежених пристроях IoT без потреби у потужних обчислювальних ядрах. Ступінь готовності рішення є високим: розроблено математичні моделі, проведено повний цикл статистичного тестування та створено програмний модуль, який слугує верифікованою базою для подальшої апаратної реалізації. Відсутність потреби у дорогих ліцензійних компонентах чи дефіцитних матеріалах додатково підвищує інвестиційну привабливість проекту.

4.2 Розрахунок витрат на здійснення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи (НДР) щодо створення апаратно-програмного комплексу потокового шифрування, під час планування, обліку і калькулювання собівартості групуються за такими статтями:

- витрати на оплату праці;
- відрахування на соціальні заходи;
- сировина та матеріали;
- витрати на комплектуючі;
- спецустаткування для наукових (експериментальних) робіт;
- програмне забезпечення для наукових (експериментальних) робіт;
- амортизація обладнання, програмних засобів та приміщень;
- паливо та енергія для науково-виробничих цілей;

– витрати на службові відрядження;

– витрати на роботи, які виконують сторонні підприємства, установи і організації;

– інші витрати;

– накладні (загальновиробничі) витрати.

Необхідно проаналізувати кожну з наведених статей.

4.2.1 Витрати на оплату праці

До такої статті витрат належать витрати на виплату основної та додаткової заробітної плати працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці [26].

Витрати на заробітну плату дослідників розраховуються на основі формули:

$$З_о = \sum_{i=1}^{k} \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.2)$$

де k – кількість посад дослідників, залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – кількість днів роботи конкретного дослідника, дн.;

T_p – середня кількість робочих днів в місяці, T_p = 22 дні.

Розрахунки витрат на заробітну плату дослідників наведено у таблиці 4.4

Таблиця 4.4 – Витрати на заробітну плату дослідників.

Посада	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Кількість днів роботи	Витрати на заробітну плату, грн
Керівник проекту	25000	1136.36	10	11363.63
Провідний спеціаліст	23000	1045.45	22	23000
Розробник	22000	1000	22	22000
Тестувальник	18000	818.18	10	8181.81
Всього				65545.44

Витрати на основну заробітну плату робітників $З_р$ за відповідними найменуваннями робіт розраховують за формулою:

$$З_{\text{р}} = \sum_{i=1}^{n} C_i \cdot t_i \,, \tag{4.3}$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника на виконання певної роботи, год.

Для визначення тарифної ставки робітника відповідного розряду $С_i$ можна скористатись формулою:

$$С_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{\text{ЗМ}}}, \tag{4.4}$$

де M_M – розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), для розрахунків прийнято M_M = 8000 грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду [26];

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об’єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середнє число робочих днів в місяці, приблизно T_p = 22 дн;

$t_{зм}$ – тривалість зміни, год.

Результати розрахунків погодинних тарифних ставок для кожного виду робіт наведено разом з розрахунками витрат в таблиці 4.5

Таблиця 4.5 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника, грн
Підготовка робочого місця	2	2	1.1	57.50	115
Установка ПЗ та середовищ розробки	4	3	1.35	70.57	282.28

Аналіз джерел та підготовка дослідження	10	7	2.2	115	1150
Розробка математичної моделі	16	7	2.2	115	1840

Продовження таблиці 4.5

Програмна реалізація алгоритмів	12	6	2	104.55	1254.6
Синтез та мінімізація логічних функцій	6	5	1.7	88.86	533.16
Статистичне тестування (NIST)	4	2	1.1	57.50	230
Всього					5405.04

Після обчислення витрат на основну заробітну плату робітників необхідно також визначити величину витрат на додаткову заробітну плату дослідників та робітників, що складає 10 … 12% від суми витрат за основну заробітну плату дослідників та робітників. Для розрахунків взято середнє значення 11%. Величину витрат на додаткову заробітну плату дослідників та робітників можна визначити за формулою:

$$З_{\text{дод}} = \left(З_{\text{о}} + З_{p}\right) \cdot \frac{Н_{\text{дод}}}{100\%} = (65545.44 + 5405.04) \cdot \frac{11}{100}\% = 7804.56 \quad (4.5)$$

4.2.2 Витрати на соціальні заходи

Відрахування на соціальні заходи включає в себе відрахування внеску на загальнообов’язкове державне соціальне страхування та для здійснення заходів щодо соціального захисту населення. Цей внесок обраховується як 22% від суми основної та додаткової заробітної плати дослідників та робітників за формулою:

$$З_{\text{Н}} = \left(З_{\text{о}} + З_{p} + З_{\text{дод}}\right) \cdot \frac{Н_{\text{ЗП}}}{100\%}, \quad (4.6)$$

Таким чином відрахування на соціальні заходи можна обрахувати як:

$$З_{\text{Н}} = (65545.44 + 5405.04 + 7804.56) \cdot \frac{22}{100}\% = 17326.11$$

де $Н_{зп}$ – норма нарахування на заробітну плату.

Таким чином, бачимо, що сума відрахувань на соціальні заходи становить 17326.11 грн.

4.2.3 Сировина та матеріали

До статті “Сировина та матеріали” належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби й предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою дослідження “Метод та засіб потокового шифрування на основі квазігруп”.

Витрати на матеріали *М* у вартісному вираженні розраховуються окремо для кожного виду матеріалів за формулою:

$$\text{М} = \sum_{j=1}^{n} \text{Н}_j \cdot \text{Ц}_j \cdot \text{К}_j - \sum_{j=1}^{n} \text{В}_j \cdot \text{Ц}_{\text{в}j}, \qquad (4.7)$$

де H_j – норма витрат матеріалу j-го найменування, кг;

n – кількість видів матеріалів;

$Ц_j$ – вартість матеріалу j-го найменування, грн/кг;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = 1,1 … 1,15);

$В_j$ – маса відходів j-го найменування, кг;

$Ц_{вj}$ – вартість відходів j-го найменування, грн/кг.

Таким чином отримані розрахунки наведено у таблиці 4.6

Таблиця 4.6 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг (од.), грн	Норма витрат, кг (шт.)	Величина відходів, кг	Ціна від ходів, грн/кг	Вартість ви траченого ма теріалу, грн
Папір офісний А4 Maestro (500 арк.)	72	2.5	0	0	198
Канцелярський набір приладдя Buromax	150	1	0	0	165
USB-накопичувач Kingston 64GB	250	1	0	0	275
Тонер для принтера HP 1010	1750	0.2	0	0	402.5
Всього					1040.5

Тоді, бачимо, що сума відрахувань на сировину та матеріали становить 1040.5 грн

4.2.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі вироби $К_в$, які використовують при дослідженні нового технічного рішення, розраховуються, згідно з їхньою номенклатурою, за формулою:

$$К_в = \sum_{j=1}^{n} Н_j \cdot Ц_j \cdot К_j, \quad (4.8)$$

де $Н_j$ – кількість комплектуючих j-го виду, шт.;

$Ц_j$ – покупна ціна комплектуючих j-го виду, грн;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ = 1,1 … 1,15).

Таким чином отримані розрахунки наведено у таблиці 4.7

Таблиця 4.7 – Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
Відлагоджувальна плата FPGA Altera Intel Cyclone V	1	8400	9240
Інтерфейсний кабель USB Cablexpert Type-A to Micro-B	1	150	165
Всього			9680

Тоді, бачимо, що сума відрахувань на комплектуючі становить 9405грн

4.2.5 Спецустаткування для наукових (експериментальних) робіт

До цієї статті належать витрати на виготовлення або придбання спеціального обладнання, стендів, пристроїв, інструментів та приладдя, необхідних для проведення наукових експериментів.

Оскільки для розробки та тестування операційного блоку потокового шифрування використовується стандартна комп’ютерна техніка та комплектуючі вироби, розраховані раніше, витрати за статтею “Спецустаткування” не передбачаються.

4.2.6 Програмне забезпечення для наукових (експериментальних) робіт

До даної статті витрат належать витрати на розробку та/або придбання спеціального програмного забезпечення, що необхідно для проведення досліджень та розробку продукту, а також витрати на проектування, формування та встановлення. До балансової вартості програмного забезпечення входять

витрати на його інсталяцію, тому ці витрати беруться додатково в розмірі 10 … 12% від вартості програмного забезпечення.

Балансова вартість програмного забезпечення обчислюється за наступною формулою:

$$В_{прг} = \sum_{i=1}^{k} Ц_{iпрг} \cdot С_{пргi} \cdot К_i, \quad (4.9)$$

де $Ц_{iпрг}$ – ціна придбання одиниці програмного засобу цього виду, грн;

$С_{пргі}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

$К_i$ – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($К_i$ = 1,10…1,12). Для обчислень візьмемо $К_i$ = 1.10;

k – кількість найменувань програмних засобів.

Таким чином, результати обчислень наведено у таблиці 4.8

Таблиця 4.8 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт.	Ціна за штуку, грн	Сума, грн
Середовище автоматизованого проектування ПЛІС Intel Quartus Prime	1	5000	5500
Інтегроване середовище розробки JetBrains PyCharm Professional	1	2000	2200
Всього			7700

Таким чином, отримано вартість програмного забезпечення, що складає 7700 грн.

4.2.7 Амортизація обладнання, програмних засобів та приміщень

До статті “Амортизація обладнання, програмних засобів та приміщень” відносять амортизаційні відрахування по кожному виду обладнання, устаткування та інших приладів і пристроїв, а також програмного забезпечення для проведення науково-дослідної роботи, за його наявності в дослідній організації або на підприємстві.

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо можуть бути розраховані з використанням прямолінійного методу амортизації за формулою:

$$\mathrm{А_{обл}} = \frac{\mathrm{Ц_б}}{\mathrm{Т_в}} \cdot \frac{t_{\mathrm{вик}}}{12}, \quad (4.10)$$

де $Ц_б$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_в$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

Розрахунки за такою статтею наведено у таблиці 4.9

Таблиця 4.9 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаціні відрахування, грн
Ноутбук ASUS	23000	3	1	638.9
Ноутбук HP	22000	3	1	611.1
Плата FPGA	8400	2	1	350
Програмне забезпечення	7700	3	1	213.8
Приміщення лабораторії досліджень	150000	30	1	416.6
Принтер	15000	2	1	625
Всього				2855.4

Таким чином, вартість амортизації обладнання, програмних засобів та приміщень складає 2855.4 грн.

4.2.8 Паливо та енергія для науково-виробничих цілей

До статті “Паливо та енергія для науково-виробничих цілей” належать витрати на придбання енергії, що витрачається з технологічною метою на проведення досліджень. Оскільки робота виконується з використанням комп'ютерної техніки та спеціалізованого обладнання, розраховуються витрати на силову електроенергію.

Витрати на силову електроенергію $B_е$ розраховують за формулою:

$$\text{В}_\text{е} = \sum_{i=1}^{n} \frac{W_{yi} \cdot t_i \cdot \text{Ц}_\text{е} \cdot \text{К}_{\text{в}ni}}{\eta_i}, \quad (4.11)$$

де W_{yt} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

$Ц_e$ – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), приймемо $Ц_e$ = 8.00 грн;

$К_{вні}$ – коефіцієнт, що враховує використання потужності, $К_{вні}$ < 1, приймемо $К_{вні}$=0.8;

η_i – коефіцієнт корисної дії обладнання, η_i < 1, приймемо η_i = 0.9.

Таблиця 4.10 – Витрати на паливо та енергію для науково-виробничих цілей

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
ПЕОМ для обчислень	0.15	176	187.73
Ноутбук HP	0.12	176	150.19
Плата FPGA	0.02	176	25.03
Принтер	0.25	5	8.8
			371.75

Таким чином, витрати на енергію для науково-виробничих цілей становлять 371.75 грн.

4.2.9 Службові відрядження

До статті “Службові відрядження” належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов’язані з проведенням випробувань машин та при ладів, а також витрати на відрядження на наукові з’їзди, конференції, наради, пов’язані з виконанням конкретних досліджень.

Витрати за статтею “Службові відрядження” розраховуються як 20…25% від суми основної заробітної плати дослідників та робітників за формулою:

$$\text{В}_\text{св} = (\text{З}_\text{О} + \text{З}_\text{Р}) \cdot \frac{\text{Н}_\text{св}}{100\%}, \quad (4.12)$$

де $H_{св}$ – норма нарахування за статтею “Службові відрядження”, прийнято $H_{св}$ = 0%.

Тоді $H_{св}$ = 0 грн, витрати за цією статтею відсутні.

4.2.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

До цієї статті належать витрати на проведення робіт, які не можуть бути виконані внутрішніми ресурсами, і для виконання яких залучаються сторонні організації.

У даній роботі до таких витрат віднесено використання спеціалізованого лабораторного обладнання партнерів університету для фінального синтезу проекту на ПЛІС (FPGA) та проведення апаратної верифікації розробленого криптографічного модуля, що вимагало специфічних ліцензійних засобів та апаратних стендів.

Витрати за статтею “Витрати на роботи, які виконують сторонні підприємства, установи і організації” розраховуються як 30…45% від суми основної заробітної плати дослідників та робітників за формулою:

$$В_{сп} = (З_О + З_Р) \cdot \frac{Н_{сп}}{100\%}, \quad (4.13)$$

де $H_{сп}$ – норма нарахування за статтею “Витрати на роботи, які виконують сторонні підприємства, установи і організації”, приймемо $H_{сп}$ = 30%.

Підставляючи необхідні значення, розрахувати норму нарахування за такою статтею можна за формулою:

$$В_{сп} = (65545.44 + 5405.04) \cdot \frac{30\%}{100\%} = 21285.14$$

Таким чином, витрати на роботи, які виконують сторонні підприємства, установи і організації становлять 21285.14 грн.

4.2.11 Інші витрати

До статті “Інші витрати” належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуються як 50…100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_{\text{в}} = (З_{\text{О}} + З_{\text{Р}}) \cdot \frac{Н_{\text{ів}}}{100\%}, \quad (4.14)$$

де $H_{ів}$ – норма нарахування за статтею “Інші витрати”, приймемо $H_{ів}$ = 50%.

Підставляючи необхідні значення, розрахувати норму нарахування за такою статтею можна за формулою:

$$I_{\text{в}} = (65545.44 + 5405.04) \cdot \frac{50\%}{100\%} = 35475.24$$

Таким чином, інші витрати становлять 35475.24 грн

4.2.12 Накладні (загальновиробничі) витрати

Накладні витрати розраховуються виходячи з основної заробітної плати дослідників та робітників, становлячи 100 … 150% від суми цих витрат. Сюди входять витрати, пов’язані з управлінням організацією, витрати на винахідництво та раціоналізацію, витрати на підготовку (перепідготовку) та навчання кадрів, витрати, пов’язані з набором робочої сили, витрати на оплату послуг банків, витрати, пов’язані з освоєнням виробництва продукції, витрати на науково-технічну інформацію та рекламу та ін.

Такі витрати обраховуються за формулою:

$$В_{\text{нзв}} = (З_{\text{О}} + З_{\text{Р}}) \cdot \frac{Н_{\text{нзв}}}{100\%} = (65545.44 + 5405.04) \cdot \frac{100\%}{100\%} = 70950.48 \quad (4.15)$$

де $H_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», прийнято $H_{нзв}$ = 100%.

Тоді, витрати на статтю, накладні (загальновиробничі) витрати, становлять 70950.48 грн.

Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат за формулою:

$$В_{\text{заг}} = З_{\text{О}} + З_{\text{Р}} + З_{\text{дод}} + З_{\text{Н}} + М + К_{\text{В}} + В_{\text{спец}} + В_{\text{прг}} + А_{\text{обл}} + В_{\text{е}} + В_{\text{св}} + В_{\text{сп}} + І_{\text{в}} + В_{\text{нзв}} \quad (4.16)$$

Підставляючи необхідні значення, розрахувати витрати на проведення науково-дослідної роботи можна за формулою:

$B_{заг}$ = (65545.44 + 5405.04 + 7804.56 + 17326.11 + 1040.5 + 9680 + 0 + 7700 + 2855.4 + 371.75 + 0 + 21285.14 + 35475.24 + 70950.48) = 245164.66 грн

Для визначення кінцевої вартості завершення науково-технічної роботи необхідно визначити етап виконання науково-технічної роботи. Дана науково технічна розробка знаходиться на останніх етапах стадії розробки промислового зразка, тому буде застосовано коефіцієнт η = 0,9. Таким чином, застосовуючи формулу отримано:

$$ЗВ = \frac{В_{заг}}{\eta} = \frac{245439.66}{0.9} = 272405.18$$

Таким чином, кінцева вартість науково-технічної роботи становить 272405.18.73 грн.

4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

Для оцінювання економічної доцільності вкладення коштів у комерціалізацію розробленого апаратно-програмного комплексу потокового шифрування на основі квазігруп необхідно розрахувати показники економічної ефективності для потенційного інвестора.

Результати дослідження проведені за темою “Метод та засіб потокового шифрування на основі квазігруп” передбачають комерціалізацію протягом 4-х років реалізації на ринку.

При розрахунку необхідно врахувати такі показники як:

ΔN – збільшення кількості споживачів пристрою, в аналізовані періоди часу, від покращення його певних характеристик. Значення відповідного показника за розрахунковими роками наведено у таблиці 4.11.

Таблиця 4.11 – Збільшення кількості споживачів пристрою, в аналізовані періоди часу

Показник	1-й рік	2-й рік	3-й рік	4-й рік
Збільшення кількості споживачів пристрою, осіб	250	400	600	800

N – кількість споживачів, які використовували аналогічний пристрій у році до впровадження результатів нової науково-технічної розробки, приймемо N = 200 осіб.

$Ц_o$ – вартість пристрою (машини, механізму) у році до впровадження результатів розробки, приймемо 5000 грн.

$\pm\Delta Ц_o$ – зміни вартості пристрою (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу, приймемо 1000 грн.

Враховуючи високу актуальність кіберзахисту для малоресурсних систем, прогнозується стійкий попит з боку інтеграторів розумних мереж та промислової автоматизації.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta П_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховується за формулою [26]:

$$\Delta П_i = (\pm\Delta Ц_o \cdot N + Ц_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \qquad (4.16)$$

де λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2025 році ставка податку на додану вартість складає 20%, а коефіцієнт = 0,8333.

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту, приймемо 30%, ρ = 0.3;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2025 році ϑ = 18%;

З наведеної формули можна розрахувати збільшення чистого прибутку для кожного з років.

Розрахунок збільшення чистого прибутку 1-го року:

$$\Delta П_1 = (1000 \cdot 200 + 5000 \cdot 250)_1 \cdot 0.8333 \cdot 0.3 \cdot \left(1 - \frac{18}{100}\right) = 297238.11 \text{ грн}$$

Розрахунок збільшення чистого прибутку 2-го року:

$$\Delta П_2 = (1000 \cdot 200 \ + 5000 \cdot 400)_2 \cdot 0.8333 \cdot 0.3 \cdot \left(1 \ - \frac{18}{100}\right) = 450981.96 \text{ грн}$$

Розрахунок збільшення чистого прибутку 3-го року:

$$\Delta П_3 = (1000 \cdot 200 \ + 5000 \cdot 600)_3 \cdot 0.8333 \cdot 0.3 \cdot \left(1 \ - \frac{18}{100}\right) = 655973.76 \text{ грн}$$

Розрахунок збільшення чистого прибутку 4-го року:

$$\Delta П_4 = (1000 \cdot 200 \ + 5000 \cdot 800)_4 \cdot 0.8333 \cdot 0.3 \cdot \left(1 \ - \frac{18}{100}\right) = 860965.56 \text{ грн}$$

Приведена вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки можна розрахувати за формулою:

$$\text{ПП} = \sum_{i=1}^{T} \frac{\Delta П_i}{(1+\tau)^i}, \qquad (4.17)$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, приймемо $\tau = 0.15$;

t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

З наведеної формули можна розрахувати приведену вартість збільшення всіх чистих прибутків на основі отриманих значень:

$$\text{ПП} = \frac{297238.11}{1.15} + \frac{450981.96}{1.15^2} + \frac{655973.76}{1.15^3} + \frac{860965.56}{1.15^4} = 1523048.32 \quad (4.17)$$

Далі необхідно розрахувати приведену вартість всіх чистих прибутків ПП за формулою:

$$PV = k_{\text{інв}} \cdot \text{ЗВ} = 1.1 \cdot 272405.18 = 299645.80, \qquad (4.18)$$

Де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{інв}$ = 1.1;

ЗВ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 272405.18 грн

Також необхідно розрахувати абсолютний економічний ефект $Е_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки за формулою:

$$Е_{абс} = ПП - PV = 1523048.32 - 299645.70 = 1223402.62, \quad (4.19)$$

де *ПП* – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, розраховано 1523048.32 грн;

PV – теперішня вартість початкових інвестицій, розраховано 299981.80 грн.

Оскільки величина $Е_{абс}$ має велике позитивне значення, це свідчить про потенційну зацікавленість інвесторів у впровадженні та комерціалізації такої науково-технічної роботи. Але для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність $Е_в$ або показник внутрішньої норми дохідності (IRR, Internal Rate of Return) вкладених інвестицій та порівняти їх з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахувати внутрішню економічну дохідність $Е_в$ можна за формулою:

$$Е_в = \sqrt[Т_ж]{1 + \frac{Е_{абс}}{PV}} - 1 = \sqrt[4]{1 + \frac{1223402.62}{299981.80}} - 1 = 0.50, \quad (4.20)$$

де $Е_{абс}$ – абсолютний економічний ефект вкладених інвестицій, розраховано 1223402.62 грн;

PV – теперішня вартість початкових інвестицій, розраховано 299981.80 грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

Розрахувати мінімальну внутрішню економічну дохідність вкладених інвестицій τ_{min} можна за формулою:

$$\tau_{min} = d + f = 0.12 + 0.3 = 0.42, \quad (4.21)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2025 році в Україні d = 0,12;

f – показник, що характеризує ризикованість вкладення інвестицій, приймемо f = 0,3.

Оскільки внутрішня економічна дохідність перевищує мінімальну, потенційний інвестор можу бути зацікавлений в даній розробці.

Далі необхідно розрахувати період окупності інвестицій $Т_{ок}$ (DPP, Discounted Payback Period), які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки за формулою:

$$Т_{ок} = \frac{1}{Е_{в}} = \frac{1}{0.50} = 2, \quad (4.22)$$

де $Е_{в}$ – внутрішня економічна дохідність вкладених інвестицій.

Таким чином, термін окупності складає 2 роки. Отриманий термін є меншим ніж три роки, це свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження цієї розробки та виведення її на ринок.

4.4 Висновки до розділу

За результатами проведеного техніко-економічного обґрунтування науково-дослідної роботи за темою «Метод та засіб потокового шифрування на основі квазігруп» встановлено, що розробка має високий комерційний потенціал, оцінений експертами у 46.67 бала. Такий результат обумовлений поєднанням високої криптографічної стійкості запропонованого методу з низькою

ресурсоємністю, що є критичною перевагою для сучасного ринку захищених систем Інтернету речей.

Детальна калькуляція витрат на реалізацію проекту показала, що загальна вартість проведення науково-дослідної роботи становить 272405.18 грн. Структура кошторису підтверджує наукоємний характер розробки, оскільки основна частка інвестицій спрямована на інтелектуальну складову — математичне моделювання, програмну реалізацію алгоритмів та верифікацію їхньої стійкості. Отримана собівартість є конкурентоспроможною та економічно виправданою, оскільки оптимізація алгоритмічної частини дозволила знизити вимоги до дороговартісного апаратного забезпечення.

Розрахунок показників економічної ефективності засвідчив високу інвестиційну привабливість проекту. Термін окупності інвестицій становить 2 роки, що є меншим за критичний бар'єр у три роки, а внутрішня норма дохідності суттєво перевищує мінімальні вимоги ринку. Це свідчить про доцільність комерціалізації розробленого апаратно-програмного комплексу та підтверджує, що впровадження результатів цієї науково-технічної роботи здатне забезпечити стабільний економічний ефект для потенційного інвестора.

ВИСНОВКИ

У цій магістерській роботі було проведено всебічне дослідження того, як можна покращити генератор випадкових чисел для шифрування даних, використовуючи спеціальні математичні структури — квазігрупи. Аналіз показав, що звичні методи захисту інформації на основі регістрів зсуву вже досягли своєї межі: вони змушують обирати між швидкістю роботи та надійністю захисту від сучасних кібератак. Тому в роботі було обрано альтернативний шлях — використання квазігруп, які завдяки своїм унікальним властивостям дозволяють створити стійкий захист, хоча раніше їх було важко ефективно масштабувати для реальних технічних завдань.

Важливим теоретичним результатом стало відкриття чіткого зв’язку між порядком квазігрупи та тим, як працює її внутрішній механізм переходів. Експерименти підтвердили гіпотезу про особливу роль простих чисел: якщо розмір квазігрупи є простим числом більше трьох, то вона автоматично утворює єдиний нерозривний ланцюжок станів. Це гарантує, що генератор працюватиме максимально довго без повторень. Натомість використання складених чисел, таких як 4 або 8, без додаткових модифікацій призводить до розпаду системи на окремі ізольовані фрагменти, що створює серйозний ризик замикання генератора на коротких циклах і знижує надійність усієї системи безпеки.

Щоб застосувати ці теоретичні знання на практиці, було розроблено вдосконалені методи побудови великих квазігруп, які підходять для сучасних стандартів шифрування. Дослідження показало, що просте поєднання менших блоків у велику структуру має межу ефективності: без додаткового перемішування система охоплює лише близько половини всіх можливих станів, що недостатньо для надійного захисту. Рішенням стало впровадження так званих ізотопічних перетворень — випадкового перемішування елементів на кожному етапі побудови. Це дозволило зруйнувати внутрішні залежності та досягти покриття майже всього простору станів, що є критично важливим показником для протидії спробам підбору ключа.

Окрему увагу в роботі було приділено тому, як зробити такий генератор економічно вигідним та ефективним для електроніки. Замість використання громіздких таблиць пам'яті було здійснено перехід до компактних логічних схем. Перебравши всі можливі варіанти малих квазігруп, вдалося знайти ідеальну структуру під ідентифікатором 11. Вона потребує мінімум ресурсів для реалізації — всього чотири логічні елементи, але при цьому забезпечує складну нелінійну залежність, необхідну для криптографії. Це вигідно відрізняє її від простіших лінійних варіантів, які легко зламати, і доводить, що висока надійність не завжди вимагає великих апаратних витрат.

Ефективність запропонованих рішень було перевірено серією стандартних статистичних тестів NIST. Виявилося, що використання лише однієї квазігрупи, навіть дуже якісної, недостатньо для повноцінного шифрування — у такій послідовності залишаються приховані закономірності. Натомість розроблена архітектура, яка поєднує дві різні квазігрупи, успішно пройшла переважну більшість перевірок, показавши високу непередбачуваність та складність згенерованих послідовностей. Техніко-економічний аналіз підтвердив, що впровадження розробленого генератора у мікросхеми дозволяє заощадити ресурси кристала та енергію порівняно з існуючими табличними методами, забезпечуючи при цьому високий рівень захисту інформації.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Rukhin A. et al. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. *NIST Special Publication 800-22 Revision 1a*. National Institute of Standards and Technology, 2010. URL: https://doi.org/10.6028/NIST.SP.800-22r1a (date of access: 10.11.2025).

2. Chaudhuri A., Maitra S., Dutta O. Design and Analysis of a New Lightweight Stream Cipher (LSC). *Progress in Cryptology – INDOCRYPT 2021*. Lecture Notes in Computer Science. 2021. Vol. 13143. P. 23–44. DOI: 10.1007/978-3-030-92062-7_2 (date of access: 12.11.2025).

3. Guel I., Ouadou M., El Hajjami S. Algebraic and Correlation Attacks on a Quasigroup-Based Stream Cipher. *International Journal of Information Security and Privacy (IJISP)*. 2022. Vol. 16, Iss. 1. P. 1–18. DOI: 10.4018/IJISP.2022010103 (date of access: 12.11.2025).

4. Bernstein D. J. ChaCha, a variant of Salsa20. *Workshop Record of SASC 2008: The State of the Art of Stream Ciphers*. 2008. P. 1–6. URL: https://cr.yp.to/chacha/chacha-20080128.pdf (date of access: 15.11.2025).

5. Canteaut A. Stream Ciphers. *Encyclopedia of Cryptography, Security and Privacy*. Springer, 2021. P. 1–4. DOI: 10.1007/978-3-642-27739-9_1489-1 (date of access: 15.11.2025).

6. Menezes A. J., Van Oorschot P. C., Vanstone S. A. *Handbook of Applied Cryptography*. CRC Press, 1996. 816 p. URL: https://cacr.uwaterloo.ca/hac/ (date of access: 15.11.2025).

7. Dubrova E. A Survey of Non-Linear Feedback Shift Registers. *IEEE Transactions on Computers*. 2011. Vol. 60, Iss. 10. P. 1–1. URL: https://ieeexplore.ieee.org/abstract/document/5730336 (date of access: 18.11.2025).

8. Courtois N. T., Meier W. Algebraic attacks on stream ciphers with linear feedback. *Advances in Cryptology – EUROCRYPT 2003*. Lecture Notes in Computer Science. 2003. Vol. 2656. P. 345–359. DOI: 10.1007/3-540-39200-9_21 (date of access: 18.11.2025).

9. Rukmani Devi, Jyoti Gupta, Chaturvedi B. K. Trends in algebraic structures and their applications in cryptography. *Mathematical Journal*. 2025. Vol. 6, Iss. 2. P. 45–52. (date of access: 20.11.2025).

10. Shcherbacov V. A. *Elements of Quasigroup Theory and Applications*. CRC Press, 2017. 598 p. DOI: 10.1201/9781315120058 (date of access: 20.11.2025).

11. Markovski S. Quasigroups and their use in cryptography. *MIPRO 2011 Proceedings of the 34th International Convention*. 2011. P. 1657–1662. URL: https://ieeexplore.ieee.org/document/5967356 (date of access: 22.11.2025).

12. Markovski S., Gligoroski D., Markovski J. Classification of Quasigroups by Random Walk on Torus. *Proceedings of the 1st International Conference on Algebraic Informatics*. 2007. P. 253–266. (date of access: 22.11.2025).

13. Gligoroski D., Markovski S., Knapskog S. J. A stream cipher based on quasigroup string transformations. *The International Conference on Near-rings and Near-fields*. 2008. P. 182–187. (date of access: 22.11.2025).

14. Gligoroski D. et al. Edon80. *ECRYPT Stream Cipher Project*. 2008.

15. Khovratovich D., Nikolić I., Rechberger C. Rotational Cryptanalysis of Edon-R. *Fast Software Encryption (FSE) 2010*. Lecture Notes in Computer Science. 2010. Vol. 6147. P. 333–346. DOI: 10.1007/978-3-642-13858-4_19 (date of access: 25.11.2025).

16. Guel I., Ouadou M., El Hajjami S. Algebraic and Correlation Attacks on a Quasigroup-Based Stream Cipher. *International Journal of Information Security and Privacy (IJISP)*. 2022. Vol. 16, Iss. 1. P. 1–18. DOI: 10.4018/IJISP.2022010103 (date of access: 12.11.2025).

17. Микитченко Б., Крайнічук (Шелепало) Г. Рекурсивний генератор з латинських квадратів / Безпека сучасних інформаційно-комунікаційних систем: матеріали Міжнародної науково-технічної конференції (SMICS-2025), 16–18 жовтня 2025 р., Львів, Україна. – Львів: ЛНУ ім. І. Франка, 2025. – С. 274–278.

18. Микитченко Б. Засіб потокового шифрування на основі латинських квадратів. Частина 1: Генератори псевдовипадкових послідовностей. *Матеріали LIV науково-технічної конференції підрозділів ВНТУ*. Вінниця, 2024. URL:

https://conferences.vntu.edu.ua/index.php/all/all2024/paper/view/20560 (date of access: 01.12.2025).

19. Shcherbacov V. A. Quasigroups in cryptology. *arXiv preprint arXiv:1007.3572*. 2018.

20. Kaur, J., Mozaffari Kermani, M. Hardware Constructions for Error Detection in Lightweight Authenticated Cipher ASCON Benchmarked on FPGA / IEEE Trans. Circuits Syst. II: Express Briefs. – 2022. – Vol. 69, Iss. 3. – P. 1234–1238. DOI: 10.1109/TCSII.2021.3136463.

21. Carlet C. *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021. 600 p. DOI: 10.1017/9781108607785 (date of access: 03.12.2025).

22. Banik S. et al. Synthesizing Quantum Circuits for Stream Ciphers: Low-Depth/Low-Qubit Implementations. *IACR Transactions on Symmetric Cryptology*. 2022. Vol. 2022, Iss. 4. P. 1–35. DOI: 10.46586/tosc.v2022.i4.1-35 (date of access: 04.12.2025).

23. Dubrova E. Algebraic Properties of Lightweight S-boxes and Their Impact on Side-Channel Resilience. *Journal of Cryptographic Engineering*. 2024. Vol. 14, Iss. 1. P. 45–58. DOI: 10.1007/s13389-023-00345-x (date of access: 04.12.2025).

24. Баришев Ю. В. Система адаптивного керування інформаційною безпекою : пат. 119265 Україна : МПК G05B 13/00. № u201613296 ; заявл. 26.12.2016 ; опубл. 25.09.2017, Бюл. № 18. URL: http://ir.lib.vntu.edu.ua/handle/123456789/18280 (date of access: 16.12.2025).

25. Лужецький В. А., Дмитришин О. В. Пристрій для шифрування даних в режимі зчеплення блоків даних : пат. 61271 Україна : МПК G06F 21/00. № u201103058 ; заявл. 15.03.2011 ; опубл. 25.07.2011, Бюл. № 14. 4 с.

26. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

ДОДАТКИ

91

Додаток А

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Назва роботи: Метод та засіб потокового шифрування на основі квазігруп. Частина 2. Генератор псевдовипадкових послідовностей

Автор роботи: Микитченко Богдан Валентинович

Тип роботи: магістерська кваліфікаційна робота

Підрозділ кафедра захисту інформації ФІТКІ , група 1 БС-24м

Коефіцієнт подібності текстових запозичень, виявлених у роботі системою StrikePlagiarism 0,8 %

Висновок щодо перевірки кваліфікаційної роботи (відмітити потрібне)

√ Запозичення, виявлені у роботі, є законними і не містять ознак плагіату, фабрикації, фальсифікації. Роботу прийняти до захисту

☐ У роботі не виявлено ознак плагіату, фабрикації, фальсифікації, але надмірна кількість текстових запозичень та/або наявність типових розрахунків не дозволяють прийняти рішення про оригінальність та самостійність її виконання. Роботу направити на доопрацювання.

☐ У роботі виявлено ознаки плагіату та/або текстових маніпуляцій як спроб укриття плагіату, фабрикації, фальсифікації, що суперечить вимогам законодавства та нормам академічної доброчесності. Робота до захисту не приймається.

Експертна комісія:

В. о. зав. кафедри ЗІ д. т. н., проф. _____ Володимир ЛУЖЕЦЬКИЙ

Гарант освітньої програми «Безпека інформаційних і комунікаційних систем» к.т.н., доцент _____ Олеся ВОЙТОВИЧ

Особа, відповідальна за перевірку _____ Валентина КАПЛУН

З висновком експертної комісії ознайомлений(-на)

Керівник _____ Галина ШЕЛЕПАЛО

Здобувач _____ Богдан МИКИТЧЕНКО

Додаток Б

КОД ПРОГРАМНОГО ЗАСОБУ

Файл: 16x16=ЛК256.ipynb

```
import random
import csv
import os
# --- Config ---
POOL_SIZE = 24
SEARCH_ATTEMPTS = 50000
SAVE_THRESHOLD = 15000000
OUT_FILE = "results_ls256.csv"
# Basic LS4 components
LS4_POOL = {
    "11": [[0, 1, 2, 3], [2, 3, 0, 1], [1, 2, 3, 0], [3, 0, 1, 2]],
    "14": [[0, 1, 2, 3], [2, 3, 1, 0], [1, 0, 3, 2], [3, 2, 0, 1]],
    "51": [[0, 2, 1, 3], [1, 3, 0, 2], [2, 1, 3, 0], [3, 0, 2, 1]],
    "54": [[0, 2, 1, 3], [1, 3, 2, 0], [2, 0, 3, 1], [3, 1, 0, 2]]
}
RAW_OPS = list(LS4_POOL.values())
def shuffle_ls(matrix):
    n = len(matrix)
    sq = [row[:] for row in matrix]
    # Rows/Cols shuffle
    random.shuffle(sq)
    sq = list(map(list, zip(*sq)))
    random.shuffle(sq)
    sq = list(map(list, zip(*sq)))
    # Symbol mapping
    mapping = list(range(n))
    random.shuffle(mapping)
    return [[mapping[val] for val in row] for row in sq]
def crossed_product(base, alpha, ops, n_base, n_ops):
    size = n_base * n_ops
    res = [[0]*size for _ in range(size)]
    for i in range(size):
        rp, rq = divmod(i, n_ops)
        for j in range(size):
            cp, cq = divmod(j, n_ops)
            op_idx = alpha[rp][cp]
            res[i][j] = base[rp][cp] * n_ops + ops[op_idx][rq][cq]
    return res
def check_cycle_fast(lk):
    n = len(lk)
    limit = n**3
    # Use bytearray for larger N to save RAM
    visited = bytearray(limit) if n > 16 else [False] * limit
    curr = random.randint(0, limit - 1)
    path_len = 0
    # Optimized bitwise ops for N=16 and N=256
    if n == 16:
        while not visited[curr]:
            visited[curr] = 1
            u0 = curr & 15
            u1 = (curr >> 4) & 15
            u2 = (curr >> 8)
            inter = lk[u2][u1]
            val_next = lk[inter][u0]
```

```
            curr = (u1 << 8) | (u0 << 4) | val_next
            path_len += 1
    elif n == 256:
        while visited[curr] == 0:
            visited[curr] = 1
            u0 = curr & 255
            u1 = (curr >> 8) & 255
            u2 = (curr >> 16)
            inter = lk[u2][u1]
            val_next = lk[inter][u0]
            curr = (u1 << 16) | (u0 << 8) | val_next
            path_len += 1

    return path_len
def append_to_csv(rank, cyc, matrix):
    with open(OUT_FILE, 'a', newline='') as f:
        writer = csv.writer(f)
        writer.writerow([f"ID_{rank}", cyc, f"{cyc/16777216:.5f}"])
        for row in matrix:
            writer.writerow(row)
        writer.writerow([])
def main():
    print(f"Starting LS256 generation. Threshold: {SAVE_THRESHOLD}")
    # 1. Prepare Elite LS16 pool
    elite_pool = []
    print("Generating base pool (LS16)...")
    while len(elite_pool) < POOL_SIZE:
        base = shuffle_ls(random.choice(RAW_OPS))
        ops = [shuffle_ls(random.choice(RAW_OPS)) for _ in range(4)]
        am = [[random.randint(0, 3) for _ in range(4)] for _ in range(4)]

        cand = crossed_product(base, am, ops, 4, 4)
        if check_cycle_fast(cand) >= 3969:
            elite_pool.append(cand)
            if len(elite_pool) % 5 == 0:
                print(f"Pool size: {len(elite_pool)}")
    print("Pool ready. Starting main search.")
    # 2. Main loop
    best_global = 0
    saved_count = 0
    for i in range(1, SEARCH_ATTEMPTS + 1):
        # Randomize components from pool
        base_16 = shuffle_ls(random.choice(elite_pool))
        ops_16 = [shuffle_ls(random.choice(elite_pool)) for _ in range(16)]
        am_16 = [[random.randint(0, 15) for _ in range(16)] for _ in range(16)]

        ls256 = crossed_product(base_16, am_16, ops_16, 16, 16)
        cyc = check_cycle_fast(ls256)

        if cyc > best_global:
            best_global = cyc
            print(f"Iter {i}: New max {cyc} ({(cyc/16777216)*100:.2f}%)")

            if cyc > SAVE_THRESHOLD:
                saved_count += 1
                append_to_csv(saved_count, cyc, ls256)
                print(f"Saved to {OUT_FILE}")

        if i % 1000 == 0:
            print(f"Progress: {i}/{SEARCH_ATTEMPTS}")
if __name__ == "__main__":
    main()
```

Файл: generator.ipynb

```
import sys
def solve_latin_square(template: list[list[int]], limit: int = None):
    """
    Генератор для знаходження рішень частково заповненого латинського квадрата.
    template: N*N матриця, де -1 позначає порожню клітинку.
    limit: Максимальна кількість рішень (None = усі можливі).
    """
    n = len(template)
    board = [row[:] for row in template]  # Робоча копія
    solutions_count = 0
    def get_empty_cell(mat):
        for r in range(n):
            for c in range(n):
                if mat[r][c] == -1:
                    return r, c
        return None
    def is_valid(mat, r, c, val):
        # Перевірка рядка
        if val in mat[r]:
            return False
        # Перевірка стовпця
        for i in range(n):
            if mat[i][c] == val:
                return False
        return True
    def backtrack():
        nonlocal solutions_count
                # Знайти наступну вільну клітинку
        empty_pos = get_empty_cell(board)
                if not empty_pos:
            # Рішення знайдено
            yield [row[:] for row in board]
```

```
            return
        row, col = empty_pos
        for num in range(n):
            if is_valid(board, row, col, num):
                board[row][col] = num
                yield from backtrack()
                # Перевірка ліміту після повернення з рекурсії
                if limit is not None and solutions_count >= limit:
                    return
                board[row][col] = -1  # Відкат
    # Запуск генератора
    for sol in backtrack():
        solutions_count += 1
        yield sol
        if limit is not None and solutions_count >= limit:
            break
# --- Блок тестування ---
if __name__ == "__main__":
    # Тест 1: Часткове заповнення (4x4)
    partial_4x4 = [[0, -1, -1, -1],
        [-1, -1,  2, -1], [-1,  2, -1,  3], [-1, -1, -1, -1]
    ]
    print(f"Solving 4x4 with partial inputs...")
    solver = solve_latin_square(partial_4x4, limit=1)
    for i, res in enumerate(solver, 1):
        print(f"Solution #{i}:")
        for row in res: print(row)
    # Тест 2: Генерація з нуля (3x3)
    empty_3x3 = [[-1]*3 for _ in range(3)]
    print(f"\nGenerating all 3x3 squares...")
    all_sols = list(solve_latin_square(empty_3x3))
    print(f"Total unique 3x3 squares found: {len(all_sols)}")
```

Файл: логічні формули для ЛК4.ipynb

```
import copy
def generate_latin_squares(n):
    board = [[None] * n for _ in range(n)]
    results = []
    def valid(r, c, v):
        for i in range(r):
            if board[i][c] == v: return False
        for j in range(c):
            if board[r][j] == v: return False
        return True
    def solve(r, c):
        if r == n:
            results.append(copy.deepcopy(board))
            return
        nr, nc = (r, c + 1) if c < n - 1 else (r + 1, 0)
        for val in range(n):
            if valid(r, c, val):
                board[r][c] = val
                solve(nr, nc)
                board[r][c] = None
    solve(0, 0)
    return results
def compute_anf_metrics(truth_table):
    # Fast Moebius Transform for ANF
    anf = list(truth_table)
    size = len(anf)
    step = 1
    while step < size:
        for i in range(0, size, step * 2):
```

```
            for j in range(i, i + step):
                anf[j + step] ^= anf[j]
        step *= 2
    # Hardware cost estimation (GE)
    gates = 0
    terms = 0
    is_linear = True
    for idx, coeff in enumerate(anf):
        if coeff:
            terms += 1
            deg = bin(idx).count('1')
            if deg > 1:
                gates += (deg - 1) # AND gates
                is_linear = False
    # XOR gates needed to sum terms
    gates += max(0, terms - 1)
    return gates, terms, is_linear
def get_cycle_period(matrix, n):
    visited = [False] * (n * n)
    max_len = 0
    for start_node in range(n * n):
        if visited[start_node]: continue
        curr = start_node
        path_len = 0
        path_set = set()
        while curr not in path_set and not visited[curr]:
            visited[curr] = True
            path_set.add(curr)
            row, col = divmod(curr, n)
            val = matrix[row][col]
            curr = val * n + row
```

```
            path_len += 1
        if path_len > max_len:
            max_len = path_len
    return max_len
def analyze_structures():
    N = 4
    squares = generate_latin_squares(N)
    print("ID,Gate_Cost,Term_Count,Max_Period,Linearity")
    for i, sq in enumerate(squares):
        # Flatten and extract bits
        flat = [sq[r][c] for r in range(N) for c in range(N)]
        # Analyze Bit 0
        tt0 = [(x >> 0) & 1 for x in flat]
        g0, t0, lin0 = compute_anf_metrics(tt0)
        # Analyze Bit 1
        tt1 = [(x >> 1) & 1 for x in flat]
        g1, t1, lin1 = compute_anf_metrics(tt1)
        total_gates = g0 + g1
        total_terms = t0 + t1
        period = get_cycle_period(sq, N)
        is_linear = "Linear" if (lin0 and lin1) else "Non-Linear"

        print(f"{i},{total_gates},{total_terms},{period},{is_linear}")
if __name__ == "__main__":
    analyze_structures()
```

Файл: Graph for 2 LS.ipynb

```
import random

import itertools

import networkx as nx

import matplotlib.pyplot as plt

from typing import List, Dict

def random_latin_square(n: int) -> list[list[int]]:

    square = [[(i + j) % n for j in range(n)] for i in range(n)]

    random.shuffle(square)

    square = list(map(list, zip(*square)))

    random.shuffle(square)

    square = list(map(list, zip(*square)))

    symbol_map = list(range(n))

    random.shuffle(symbol_map)

    return [[symbol_map[val] for val in row] for row in square]

def generate_sequence_two_squares(l1: list[list[int]], l2: list[list[int]],
initial_triplet: list[int]) -> list[int]:

    if len(initial_triplet) != 3: raise ValueError("Start triplet must be len
3")

    sequence = list(initial_triplet)

    used_triplets = {tuple(initial_triplet)}

    while True:

        u3, u2, u1 = sequence[-3], sequence[-2], sequence[-1]

        try:

            val = l2[l1[u1][u2]][u3]

        except IndexError: break

        sequence.append(val)

        new_triplet = (sequence[-3], sequence[-2], sequence[-1])

        if new_triplet in used_triplets: break

        used_triplets.add(new_triplet)

    return sequence[:-2]
```

```
def visualize_state_graph(l1: list[list[int]], l2: list[list[int]], title: str):
    n = len(l1)
    if n > 5:
        print(f"Skipping visualization for N={n} (too many nodes)")
        return
    G = nx.DiGraph()
    states = list(itertools.product(range(n), repeat=3))
    G.add_nodes_from(states)
    for u3, u2, u1 in states:
        nxt = l2[l1[u1][u2]][u3]
        G.add_edge((u3, u2, u1), (u2, u1, nxt))
    components = list(nx.weakly_connected_components(G))
    print(f"Components: {len(components)}")
    plt.figure(figsize=(10, 10))
    nx.draw(G, nx.spring_layout(G, seed=42), node_size=50, arrowsize=8,
alpha=0.6)
    plt.title(title)
    plt.savefig("graph_state.png")
    plt.show()
def crossed_product(lk_p: List[List[int]], ops_sigma: Dict[str,
List[List[int]]], map_alpha: List[List[str]]) -> List[List[int]]:
    ord_p, ord_q = len(lk_p), len(next(iter(ops_sigma.values())))
    new_ord = ord_p * ord_q
    res = [[0] * new_ord for _ in range(new_ord)]
    for i in range(new_ord):
        for j in range(new_ord):
            p, a = divmod(i, ord_q)
            q, b = divmod(j, ord_q)
            val_p = lk_p[p][q]
            op_key = map_alpha[p][q]
            val_a = ops_sigma[op_key][a][b]
```

```
            res[i][j] = val_p * ord_q + val_a

    return res

def generate_cp_square(p_size, q_size):

    # Generates a CP square of size P*Q using random components

    lk_p = random_latin_square(p_size)

    ops_sigma = {f'op{i}': random_latin_square(q_size) for i in range(q_size)}

    op_keys = list(ops_sigma.keys())

    map_alpha = [[random.choice(op_keys) for _ in range(p_size)] for _ in
range(p_size)]

    return crossed_product(lk_p, ops_sigma, map_alpha)

if __name__ == "__main__":

    N_TARGET = 16 # Target size

    P, Q = 4, 4   # Components (4x4 = 16)

    print(f"Generating two Latin Squares of order {N_TARGET} using Crossed
Product ({P}x{Q})...")

    # 1. Generate LK1

    lk1 = generate_cp_square(P, Q)

    print(f"LK1 Generated. Size: {len(lk1)}x{len(lk1)}")

    # 2. Generate LK2

    lk2 = generate_cp_square(P, Q)

    print(f"LK2 Generated. Size: {len(lk2)}x{len(lk2)}")

    # 3. Generate Sequence

    start_node = random.sample(range(N_TARGET), 3)

    print(f"Start Triplet: {start_node}")

    seq = generate_sequence_two_squares(lk1, lk2, start_node)

    print(f"Sequence Length: {len(seq)}")

    print(f"Sequence Preview: {seq[:50]}...")

    # Visualization only if N is small

    if N_TARGET <= 4:

        visualize_state_graph(lk1, lk2, f"States N={N_TARGET}")
```

ІЛЮСТРАТИВНА ЧАСТИНА

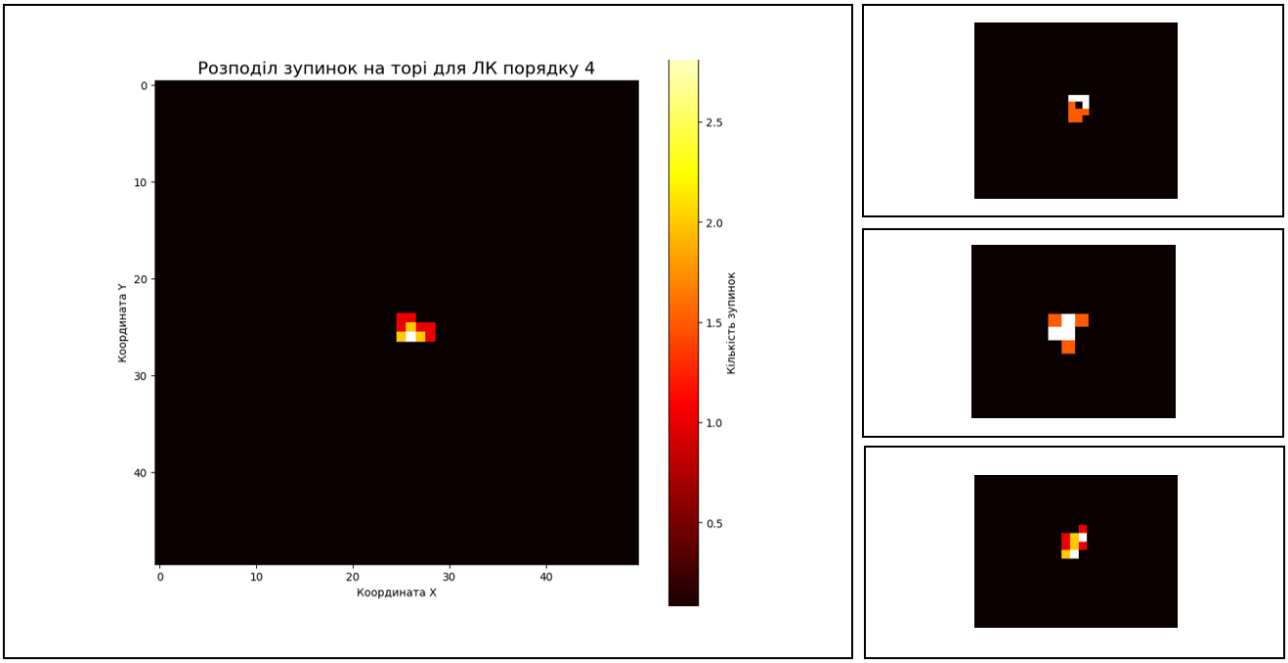
МЕТОД ТА ЗАСІБ ПОТОКОВОГО ШИФРУВАННЯ НА ОСНОВІ КВАЗІГРУП. ЧАСТИНА 1. ГЕНЕРАТОР ПСЕВДОВИПАДКОВИХ ЧИСЕЛ НА ОСНОВІ ОПЕРАЦІЙ З КВАЗІГРУПАМИ.

Виконав: студент групи 1БС-24м
спеціальності 125 Кібербезпека та захист інформації
_____ Богдан МИКИТЧЕНКО
16 грудня 2025 р.
Керівник: к.ф.-м.н., доцент каф. ЗІ
_____ Галина ШЕЛЕПАЛО
16 грудня 2025 р.

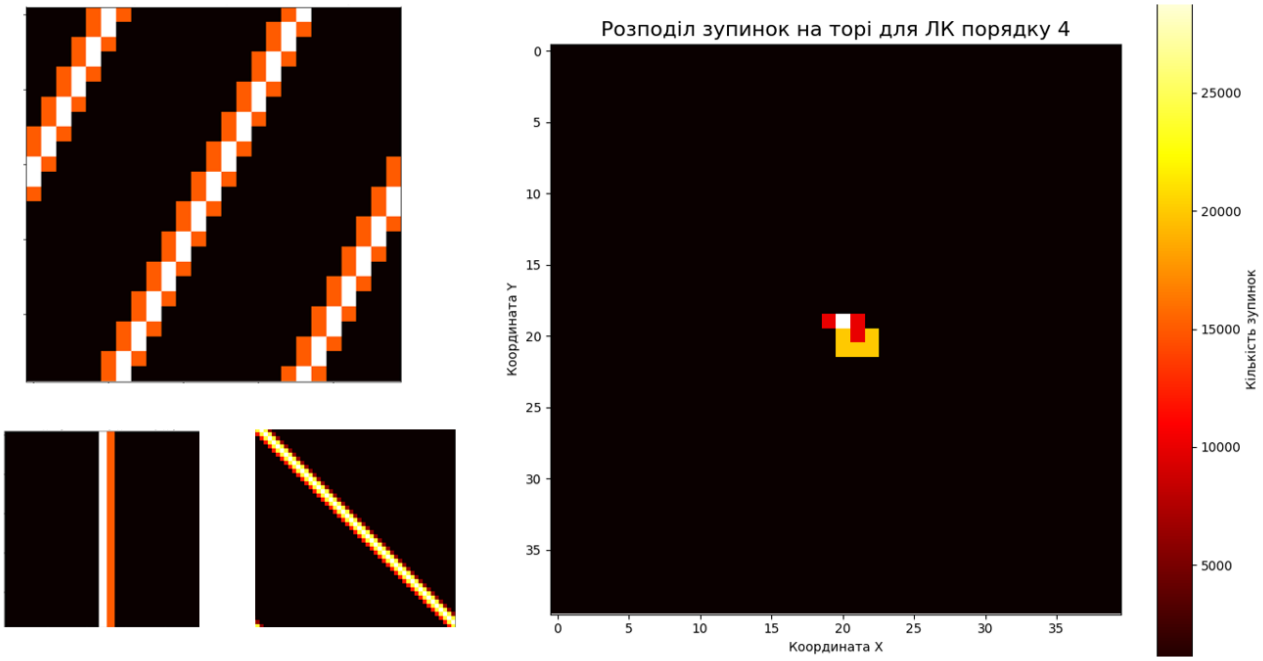
Зведені результати аналізу графа станів для просторів N^2 та N^3.

Простір станів	Порядок N	Тип числа	Макс. довжина ПВП	«Чемпіонська» структура графа
N^2	N=4	Складене	**N^2**	[N^2-1, 1]
N^2	N=5	**Просте > 3**	**N^2+1**	**[N^2]**
N^2	N=6	Складене	**N^2**	[N^2-1, 1]
N^2	N=7	**Просте > 3**	**N^2+1**	**[N^2]**
N^3	N=4	Складене	**N^3**	[N^3-1, 1]
N^3	N=5	**Просте > 3**	**N^3+1**	**[N^3]**
N^3	N=6	Складене	**N^3**	[N^3-1, 1]
N^3	N=7	**Просте > 3**	**N^3+1**	**[N^3]**

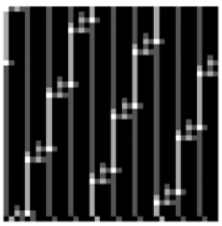
Приклади розподілу відвідувань на торі для коротких послідовностей (*N*=4)



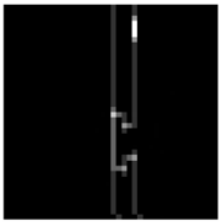
Патерни розподілу для масштабованих послідовностей (*N*=4). Найбільший рисунок – аномалія.



Порівняння методів та їх результатів



QG4-1



QG4-2

QG4-5

0 1 2 3
1 0 3 2
2 3 0 1
3 2 1 0

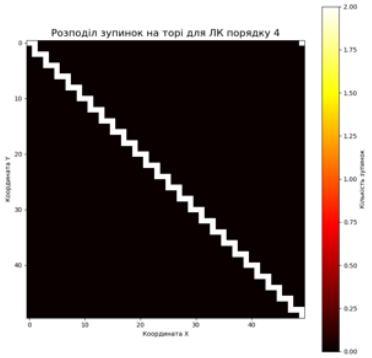
QG4-1

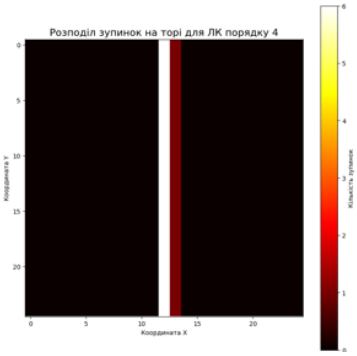
0 1 2 3
1 0 3 2
2 3 1 0
3 2 0 1

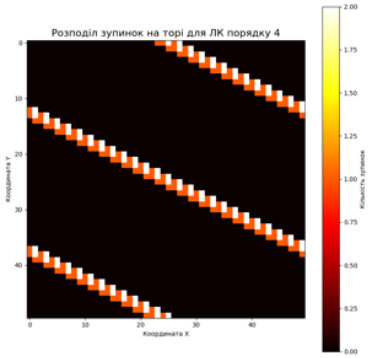
QG4-2

0 1 2 3
1 2 3 0
2 3 0 1
3 0 1 2

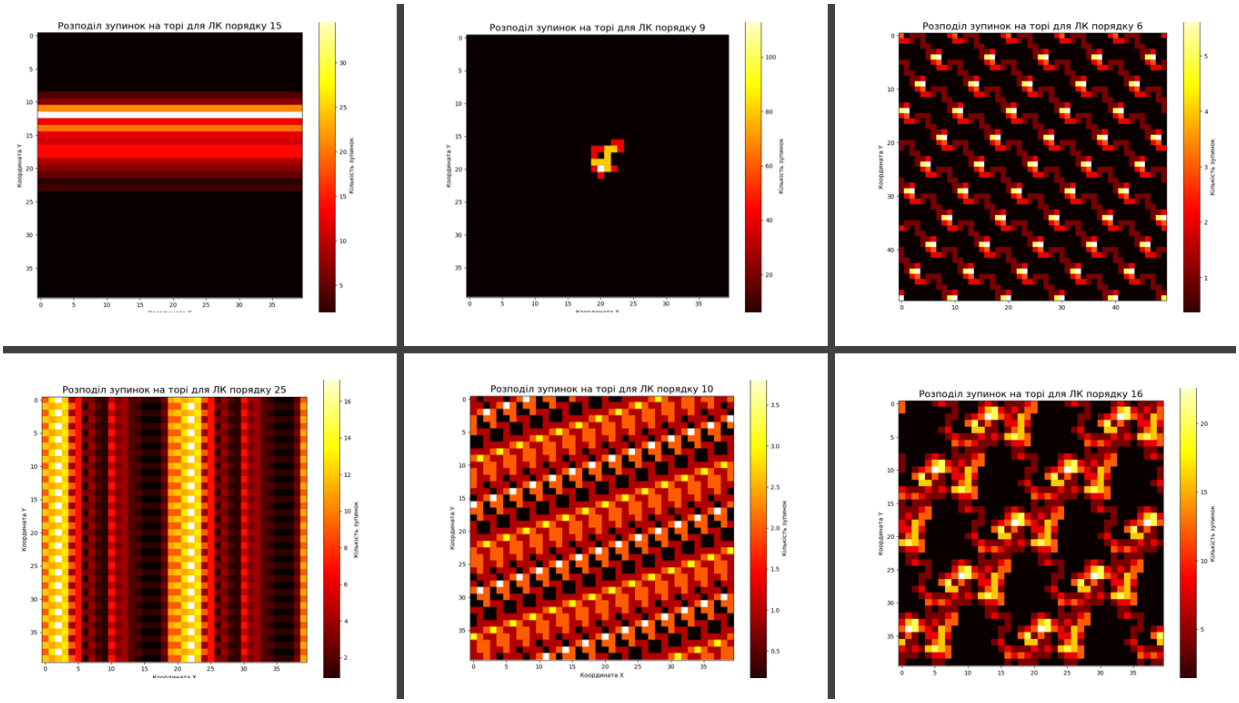
QG4-5



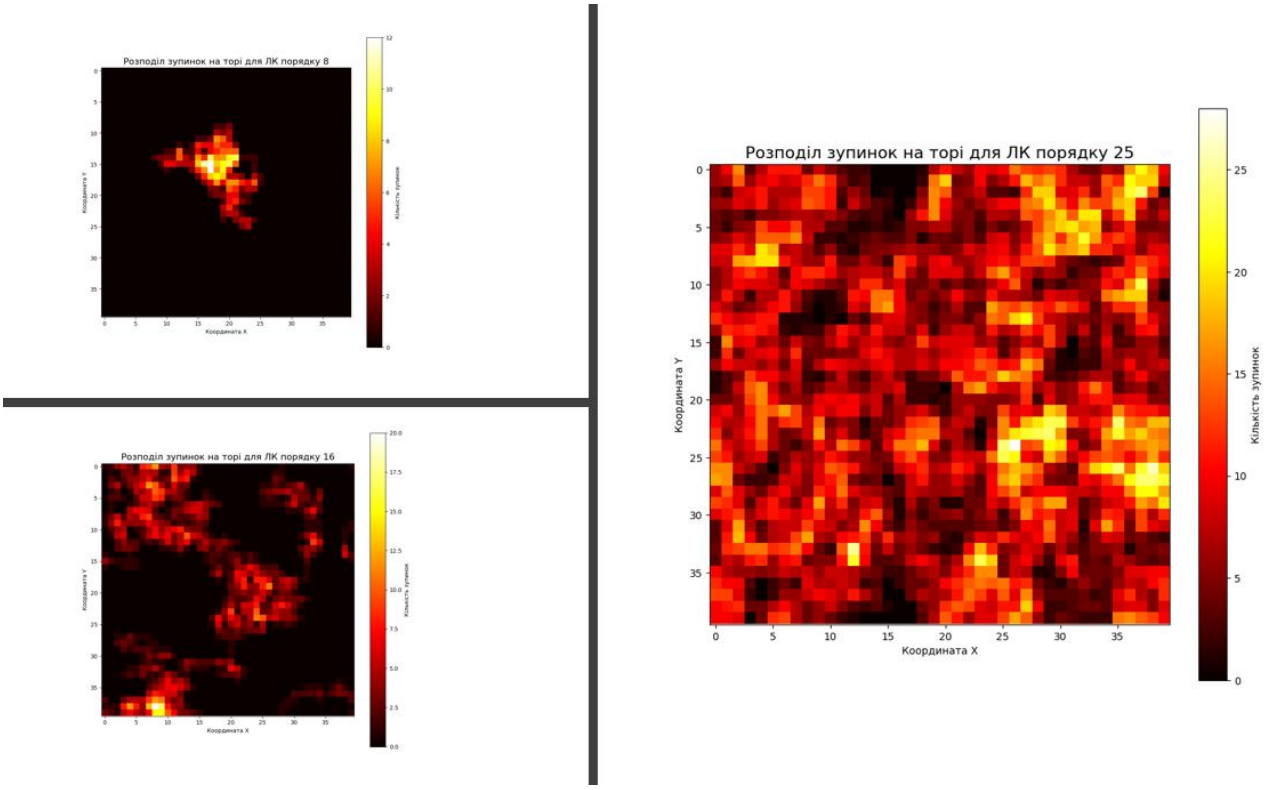




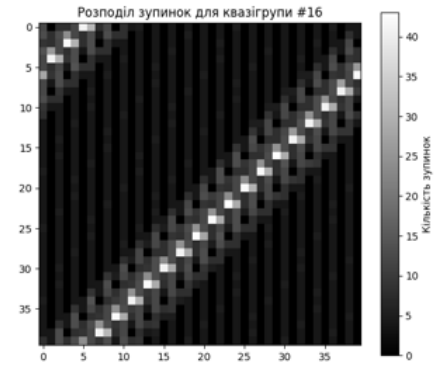
Візуалізація траєкторій на торі для латинських квадратів різних порядків ($N = 6–25$)

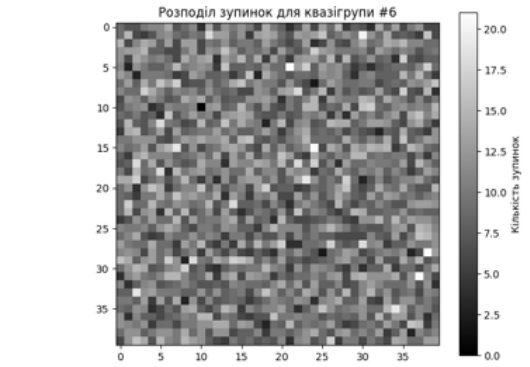


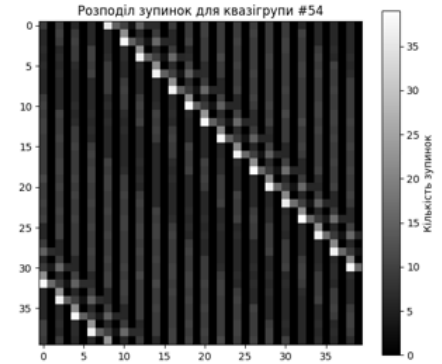
Візуалізація розподілу відвідувань на торі для генератора на основі двох ЛК без циклічного повторення для *N*=8, 16, 25.

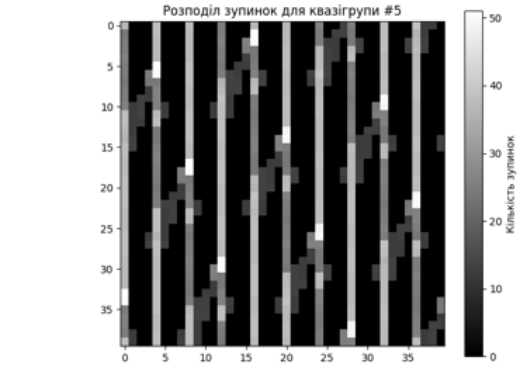


Відтворені результати E-перетворень на торі

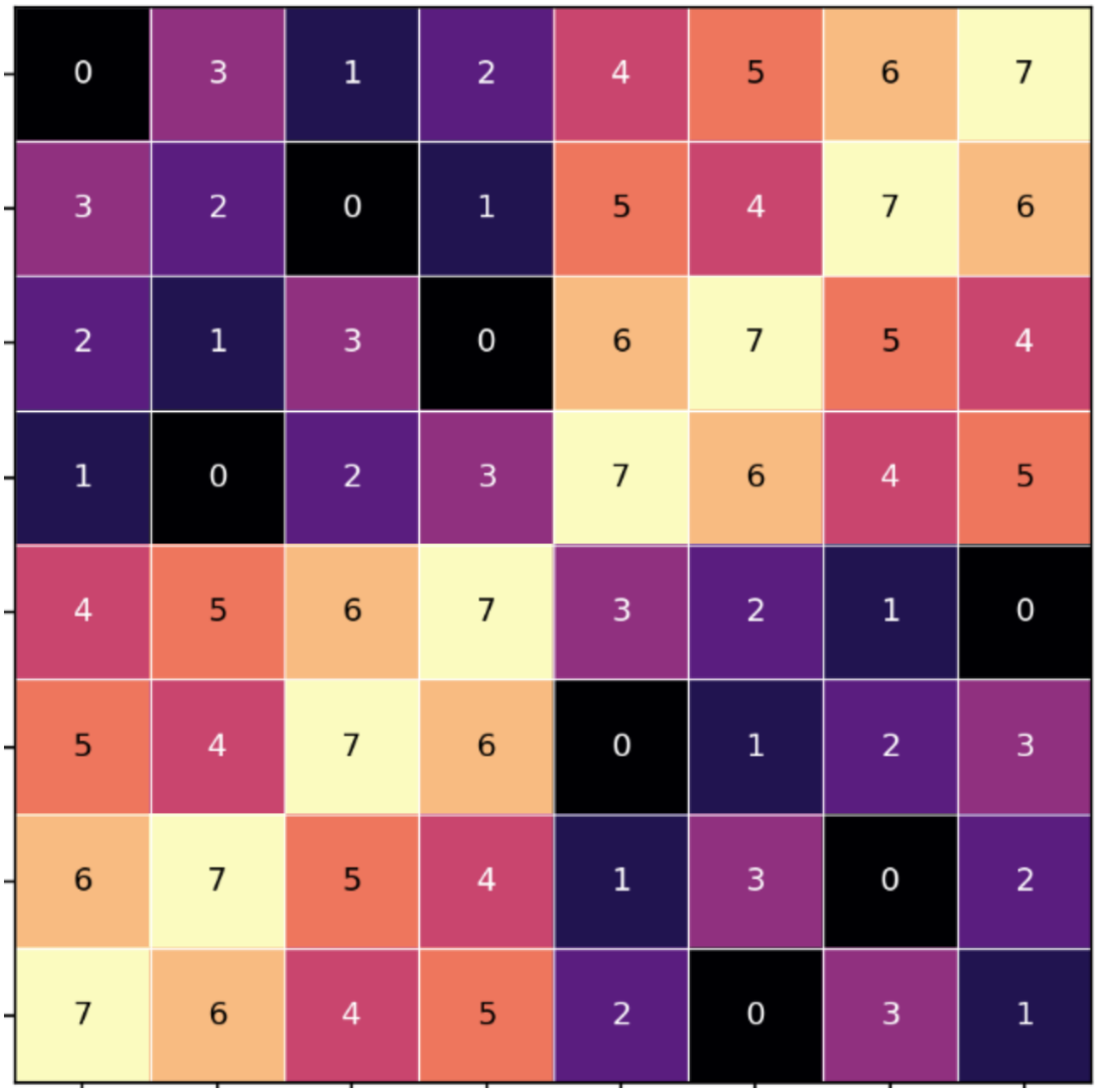




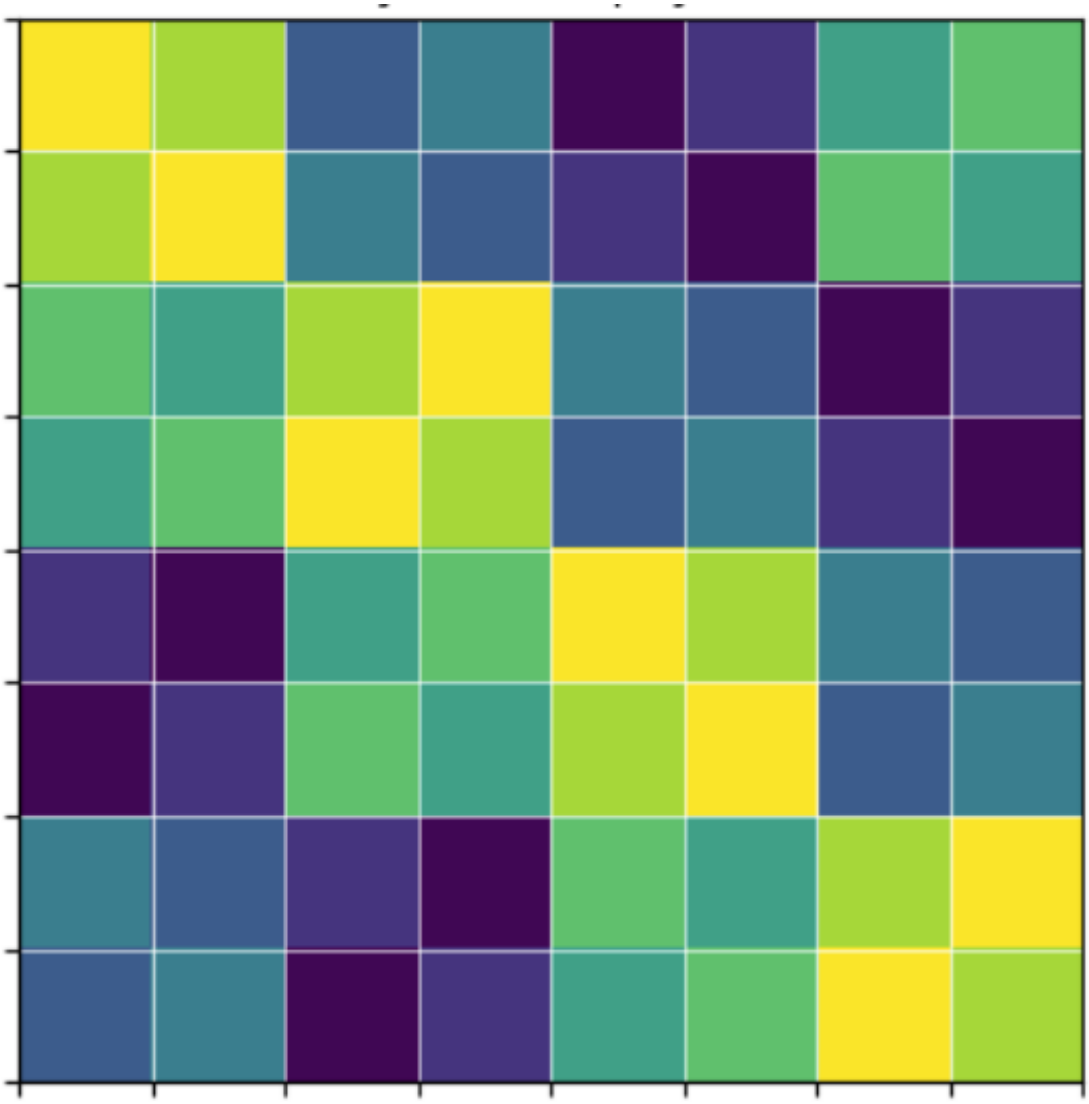




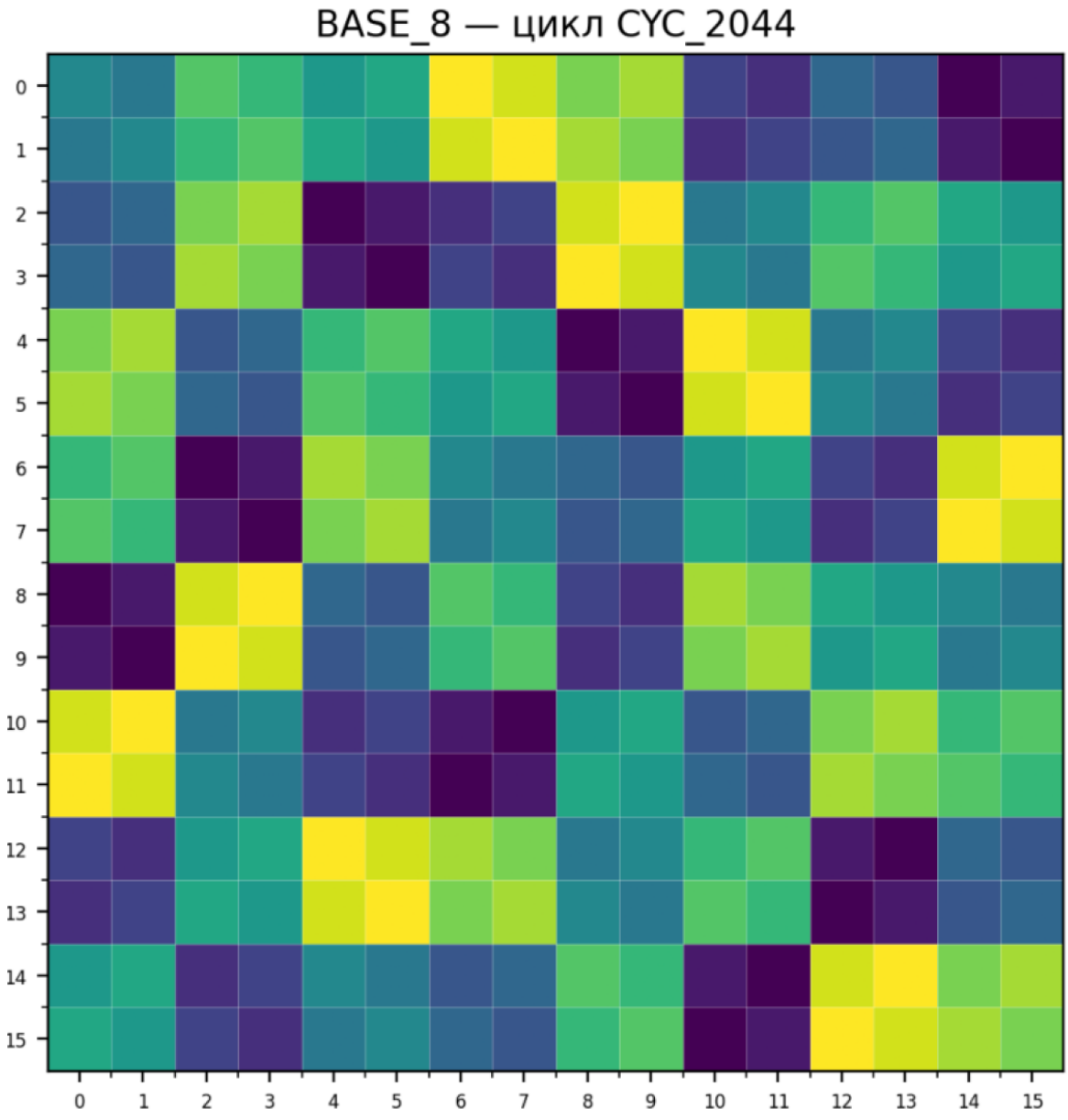
ЛК8 від 2 × 4 з блоковою симетрією (не комутативний)



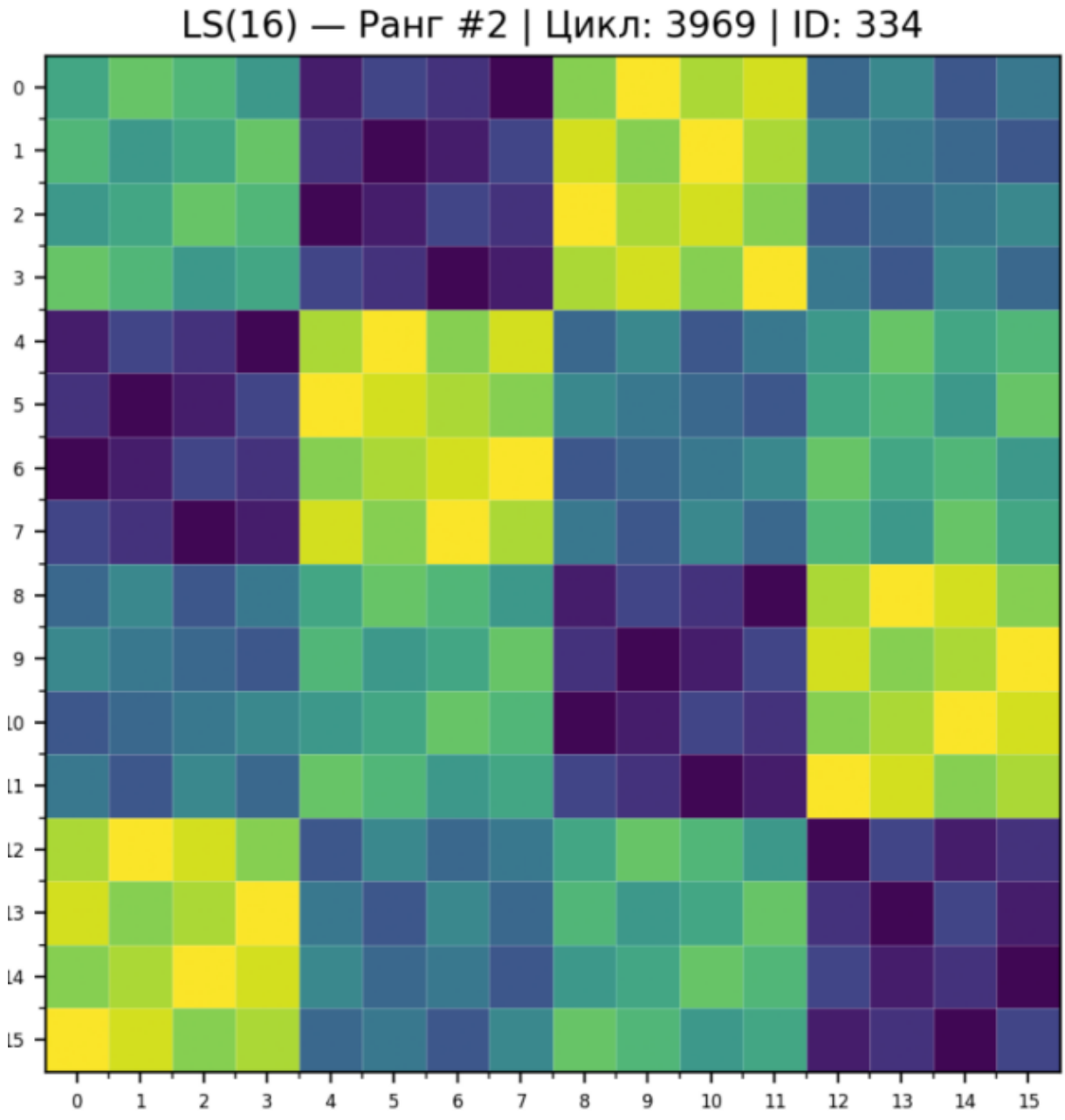
ЛК8 від 4 × 2 з візуальною симетрією (не комутативний)



ЛК16 – візуально не симетричний



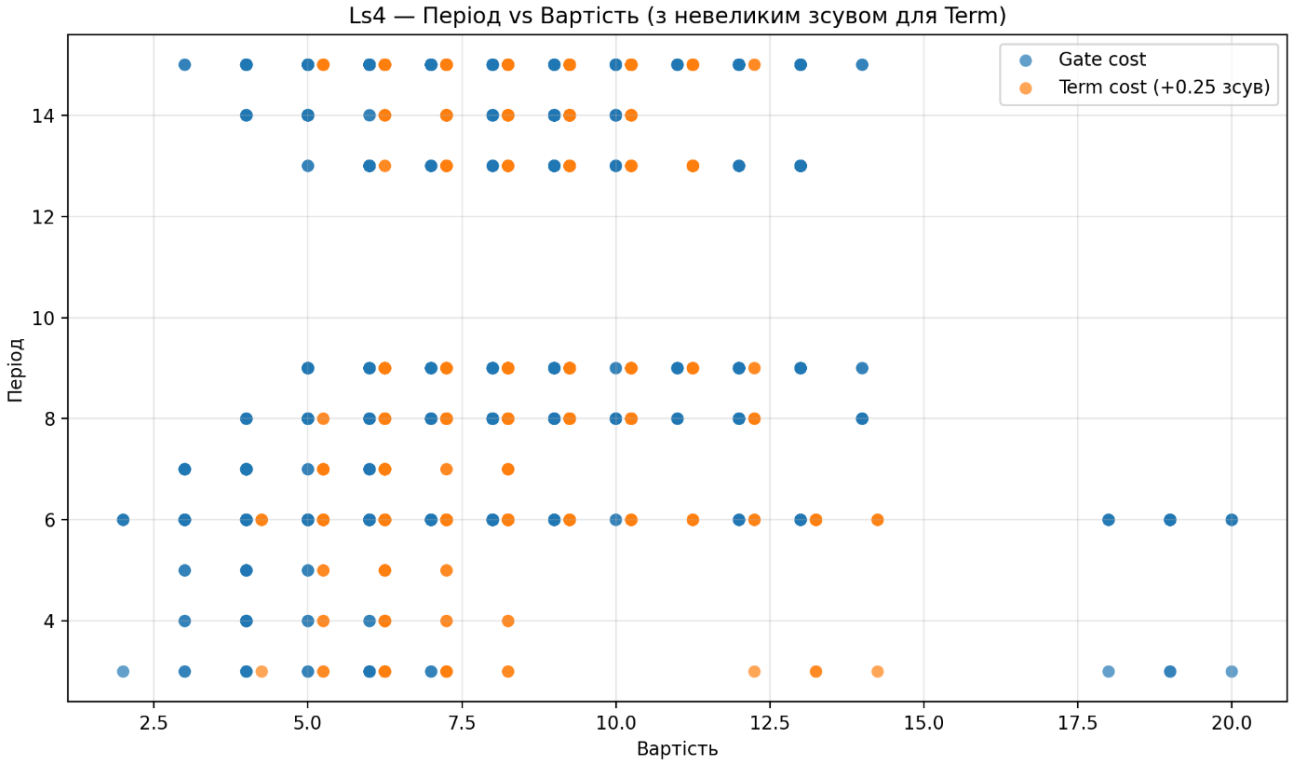
ЛК16 з 4×4


LS(16) — Ранг #2 | Цикл: 3969 | ID: 334
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

Порівняльні характеристики квазігруп різних порядків

Порядок N	Схема	Простір станів N^3	Макс. цикл	Покриття, %	Компоненти
16	8×2	4 096	2 044	49,9	Ізотопи ЛК(8) + ЛК(2)
16	4×4	4 096	3 969	96,9	Ізотопи ЛК(4) + ЛК(4)
64	8×8 (чистий)	262 144	16	0,006	Z_8 без модифікацій
64	8×8 (ізотопи)	262 144	261 121	99,61	Ізотопи Z_8
256	16×16	16 777 216	4 682 240	27,91	Ізотопи ЛК(16)

Залежність періоду генератора від апаратної складності.



Порівняння усереднених показників P-value для основних конфігурацій

Тест NIST	Gen2 (один ЛК)	Gen3 (композитний)	Gen3 (ізотопний ланцюг)
Frequency (Monobit)	0,980	0,530	0,650
Block Frequency	0,420	0,610	0,580
Runs Test	0,350	0,450	0,710
Longest Run	0,550	0,680	0,620
Linear Complexity	0,930	0,190	0,640
Approximate Entropy	<0,010	0,260	0,970
Overlapping Template	<0,010	0,750	0,350
Serial Test	<0,010	0,230	0,590
Spectral (DFT)	<0,010	<0,010	<0,010
Загальна оцінка	Незадовільно	Задовільно	Відмінно

Порівняння проходження тестів NIST для різних генераторів

Генератор	Пройдено тестів	Провалено тестів	Критичні провали	Approximate Entropy	Linear Complexity
Розроблений (Gen3, Isotopic Chain)	14/15	1/15	DFT	0,970	0,640
ChaCha20	15/15	0/15	—	>0,99	>0,99
AES-CTR	15/15	0/15	—	>0,99	>0,99
HC-128	15/15	0/15	—	>0,99	>0,99
Trivium	15/15	0/15	—	>0,99	>0,99

Схема роботи алгоритму