



Ю. Ю. Білас, В. В. Різник

Національний університет "Львівська політехніка", м. Львів, Україна

АНАЛІЗ ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ МОНОЛІТНИХ ІКВ-КОДІВ ПОРІВНЯНО З КОДАМИ РІДА-СОЛОМОНА

У сучасних системах зв'язку та оброблення інформації надзвичайно важливо забезпечити ефективне та завадостійке кодування даних, особливо в умовах підвищеного рівня шуму та обмежених ресурсів. Досліджено ефективність кодів на підставі монолітних ідеальних кільцевих в'язанок порівняно з кодами Ріда-Соломона для кодування та передавання даних у зашумлених каналах зв'язку. Розроблено програмний застосунок для моделювання процедури передавання даних через канал зв'язку з контрольованими параметрами завад, яка дає змогу оцінити витрати на оброблення, ефективність виправлення помилок, рівень надлишковості та енергоспоживання різних методів кодування. Використано мову C# та технологію WPF (англ. *Windows Presentation Foundation*). Унаслідок проведеного експерименту встановлено, що коди на підставі в'язанок забезпечують значну ефективність виправлення помилок, особливо в умовах значних завад. Досліджено, що, порівняно з кодами Ріда-Соломона, в'язанки демонструють меншу надлишковість та покращену енергоефективність завдяки зниженню частоти зміни бітів у повідомленнях, що передаються (приблизно у 3,5 рази менша частота). Водночас, з'ясовано, одним із недоліків кодів на підставі монолітних ідеальних кільцевих в'язанок є значно більші обчислювальні витрати, що може обмежувати їх застосування у високошвидкісних мережах із жорсткими вимогами до затримки. Аналізом результатів підтверджено, що у середовищах із високим рівнем завад в'язанки мають значні переваги, оскільки дають можливість досягти більшої частки успішних передавань порівняно з традиційними методами. Також продемонстровано, що в'язанки, порівняно з кодами Ріда-Соломона, мають цікаву властивість – вони інколи можуть виправляти дуже велику кількість помилок, жертвуючи детермінованістю кількості виправлених помилок. Це також додає їм переваги у середовищах, де обсяги даних величезні, а завади істотні. Усе перераховане вище робить їх перспективними для використання в мобільних і супутникових комунікаційних системах, а також у пристроях Інтернету речей, де енергоефективність відіграє критично важливу роль. У перспективі подальші дослідження можна спрямувати на удосконалення алгоритмів кодування монолітних ідеальних кільцевих в'язанок для зменшення обчислювальної складності та підвищення швидкодії, а також на вивчення особливостей їхнього застосування в нових технологіях бездротового зв'язку та космічних комунікацій.

Ключові слова: ідеальні кільцеві в'язанки; кодування; завадостійкість; енергоефективність.

Вступ / Introduction

Обсяг даних у сучасному світі зростає з неймовірною швидкістю. Кодування даних відіграє ключову роль у цих процесах, даючи змогу оптимізувати роботу систем і зменшувати витрати на передачу даних. У каналах зв'язку, де сигнали можуть спотворюватися шумом, для запобігання втраті даних використовуються методи завадостійкого їх кодування, які додають надлишкову інформацію для виправлення помилок. Однак завадостійкість не є безкоштовною – вона досягається завдяки більшому використанню ресурсів для безпосередньої роботи з даними.

Значний внесок у розвиток теорії завадостійкого кодування зробили такі науковці, як Р. Хеммінг [7], який запропонував клас коригувальних кодів, і К. Шеннон [19], який заклав теоретичні основи інформаційної теорії. Подальші дослідження привели до розроблення де-що потужніших методів, таких як коди [5], а також Тур-

бо-коди [2], що забезпечили значне підвищення ефективності передачі даних у зашумлених середовищах.

Попри значний прогрес, питання пошуку нових схем кодування, які забезпечували б високий рівень коригування помилок за мінімального збільшення надлишковості, залишається відкритим. Одним із перспективних напрямів є використання ідеальних кільцевих в'язанок (ІКВ-кодів), запропонованих у працях українських науковців [11]. Ці структури, зокрема монолітні ІКВ-коди (далі – МІКВ-коди) дають можливість ефективно виправляти помилки у великих кількостях, зберігаючи значну енергоефективність, що є особливо важливим для мобільних і супутникових систем зв'язку. ІКВ-коди відрізняються своєю можливістю виявляти та виправляти багато помилок, не додаючи при цьому надто великої надмірності, а також зберігаючи значну енергоефективність, що є особливо важливим для мобільних і супутникових систем зв'язку.

Інформація про авторів:

Білас Юрій Юрійович, аспірант, кафедра автоматизованих систем управління. Email: yurii.y.bilas@lpnu.ua;

<https://orcid.org/0009-0006-0051-5081>

Різник Володимир Васильович, д-р техн. наук, професор, кафедра автоматизованих систем управління.

Email: volodymyr.v.riznyk@lpnu.ua; <https://orcid.org/0000-0002-3880-4595>

Цитування за ДСТУ: Білас Ю. Ю., Різник В. В. Аналіз ефективності використання монолітних ІКВ-кодів порівняно з кодами Ріда-Соломона. Науковий вісник НЛТУ України. 2025, т. 35, № 1. С. 127–136.

Citation APA: Bilas, Yu. Yu., & Riznyk, V. V. (2025). Analysis of the efficiency of using monolithic IRB codes in comparison with Reed-Solomon codes. *Scientific Bulletin of UNFU*, 35(1), 127–136. <https://doi.org/10.36930/40350116>

На тему ІКВ-кодів було проведено низку досліджень, проте вони часто зосереджуються на методах їх побудови [11, 14, 22], на їх застосуванні [11, 13], а також тільки на аналітичному порівнянні ІКВ-кодів з іншими методами кодування [12, 14]. Наразі ці математичні структури ще не є достатньо вивчені, зокрема досі не проведено достатньо емпіричних порівняльних досліджень МКВ-кодів та інших завадостійких кодів.

Актуальність дослідження полягає в тому, що емпіричне порівняння характеристик кодів МКВ та інших способів кодування дає змогу визначити доцільність їхнього використання у каналах зв'язку із завадами.

Завданням є розроблення програмного застосунок для моделювання процедури передавання даних каналами зв'язку, на який діють завади, використовуючи коди МКВ, коди Ріда-Соломона, як окремого випадку кодів БЧХ (Боуза-Чоудхурі-Хоквінгема, англ. *Bose-Chaudhuri-Hocquenghem Codes*, *BCH codes*), звичайне двійкове подання числа, а також подальший аналіз отриманих характеристик використаних кодів і їх порівняння.

Об'єкт дослідження – використання монолітних ідеальних кільцевих в'язанок (МКВ) для кодування даних.

Предмет дослідження – методи і засоби порівняння характеристик МКВ-кодів порівняно з кодами Ріда-Соломона, що дасть змогу визначити доцільність їхнього використання у каналах зв'язку із завадами.

Мета роботи – проаналізувати ефективність використання монолітних ІКВ-кодів порівняно з кодами Ріда-Соломона, що дасть змогу визначити їхні переваги та недоліки під час передачі даних каналами зв'язку.

Для досягнення зазначеної мети визначено такі *основні завдання дослідження*:

- спроектувати та розробити програмний застосунок для моделювання процедури пересилання даних каналами зв'язку із завадами, використовуючи різні методи кодування даних, зокрема коди МКВ, коди Ріда-Соломона та звичайні двійкові подання чисел, що дасть змогу емпірично дослідити їхні властивості;
- провести експеримент з різноманітними параметрами кодів, що дасть можливість отримати дані, що описують ефективність застосування кожного з кодів у різноманітних ситуаціях;
- проаналізувати результати проведеного експерименту і зробити висновки стосовно переваг та недоліків кодів МКВ порівняно з кодами Ріда-Соломона, що забезпечує краще розуміння того, який з методів доцільно обирати для вирішення певних проблем.

Аналіз останніх досліджень та публікацій. Сучасна телекомунікаційна інфраструктура наразі споживає близько 1 % світової електроенергії, і це споживання зростає [20]. Дослідження енергоефективності в телекомунікаціях зараз знаходяться в центрі уваги багатьох лабораторій і консорціумів по всьому світу. Понад це, автори припускають, що якщо наші комунікаційні технології не будуть постійно покращувати, то енергоспоживання мереж може досягнути страхітливих 10 %. Це підштовхує до думки, що енергоефективність процедури передачі даних є дуже важливою для каналу зв'язку, то використання кодів зі значною енергоефективністю могло б покращити ситуацію, зекономивши ресурси та допомогти екології планети.

Існують дослідження, які доводять, що часті перемикування бітів не є енергоефективними. Наприклад, у роботі [3] автор встановив, що в елементах пам'яті РСМ (англ. *Pulse Code Modulation*) часті перемикування бітів

споживають більше енергії та більше зношують пристрій. Елементи пам'яті також беруть участь у пристроях надсилання, передачі, оброблення даних, тож зменшення частоти перемикування бітів у закодованих даних покращує енергоефективність системи, зменшуючи енерговитрати через перемикування бітів в елементах пам'яті.

Іншим прикладом є робота [6], в якій автори звертаються до проблеми залежності збільшення енергоспоживання GPU CUDA зі збільшенням частоти зміни бітів. Вони вказують на те, що найбільші втрати в цій сфері пов'язані зі значною витратою енергії на перемикування бітів. У своїй роботі автори запропонували підхід, за допомогою якого можна зменшити кількість перемикувань бітів, не зменшивши при цьому швидкодюю, і зекономити близько 9,3 % енергії. У випадках, коли системи використовують графічні процесори і потребують присутності завадостійких характеристик, зменшення перемикування бітів у даних зменшить витрати енергії.

У роботі [10] автори, на прикладі застосування алгеброгеометричних кодів, довели що для деяких кодів збільшення завадостійкості збільшує енергетичний виграш під час передачі даних. Зважаючи на це, коди МКВ, які здатні виправляти велику кількість помилок, здатні також збільшити енергетичний виграш при надсиланні повідомлень.

У роботі [4] автори провели перевірку ефективності сигнально-кодових конструкцій завадостійких кодів методом математичного моделювання для різних телекомунікаційних каналів шляхом визначення енергетичного виграшу кодування, і отримали результати, які свідчать про більшу енергоефективність кодів Ріда-Соломона над більшістю кодів, коли частота виникнення помилок на біт є дуже значною. Це свідчить про те, що у сильно зашумлених каналах зв'язку коди Ріда-Соломона є кращими за інші (Turbo, Polar, LDPC, Convolutional).

Ідеальні кільцеві в'язанки (ІКВ-коди) – це послідовності, елементами яких можуть бути цілі додатні числа або впорядковані цілочислові t -набори [11]. Одновимірні ІКВ-коди – це упорядкована послідовність $K_n = (k_1, k_2, \dots, k_i, \dots, k_n)$ чисел, на якій всі можливі кільцеві суми вичерпують значення чисел натурального ряду від 1 до $S_n = n(n-1)/R+1$, де n – кількість кільцевих сум з однаковими числовими значеннями. З'ясовано, що побудовані на підставі ІКВ-кодів завадостійкі коди набувають здатності виявляти до $n-1$ або виправляти до $n/2-1$ помилок для парних і виявляти до n або виправляти до $(n-1)/2$ помилок для непарних значень n . Наведено приклади для 31-розрядних ($S_n = 31$) оптимізованих циклічних ІКВ-кодів, які дають змогу виявляти до 15 і виправляти до 7 помилок, прямуючи зі зростанням числа розрядів до 50 і 25 % відповідно від довжини коду. Як було зазначено раніше, особливу цікавість становить використання монолітних ІКВ-кодів. Під монолітним розуміємо код, дозволені комбінації якого складаються з двох різномісних пакетів однойменних символів, що входять один поруч одного. Якщо множина ваг двійкових розрядів, разом з множиною сум поряд розміщених вагових розрядів, вичерпує значення чисел натурального ряду від 1 до суми усіх двійкових ваг, це – монолітний ІКВ-код.

Попри значний прогрес у дослідженні ІКВ-кодів, більшість наявних робіт зосереджуються на їх побудові та аналітичному порівнянні з іншими методами кодування, тоді як емпіричних досліджень їхньої ефектив-

ності в умовах завад недостатньо. Це створює наукову прогалину, яку важливо заповнити для визначення доцільності використання МКВ-кодів у сучасних комунікаційних системах. Отже, проведення цього дослідження є актуальним, оскільки дає змогу не тільки підтвердити теоретичні припущення, а й надати практичні рекомендації щодо застосування цих кодів у реальних умовах.

Основні переваги монолітного ІКВ-коду порівняно із циклічним – вища ефективність оброблення багатовимірних масивів даних, можливість захисту інформації від несанкціонованого доступу, швидке виправлення помилок [16].

Матеріали та методи дослідження. Використано метод аналізу та узагальнення для аналізу наявних методів кодування та їх властивостей. Дослідження базується на теоретичних положеннях теорії інформації та теорії кодування, зокрема аналізі коригувальних властивостей кодів і їхнього впливу на надмірність та енергоспоживання. Оцінювання ефективності кодів на

підставі монолітних ідеальних кільцевих в'язанок проводилося шляхом комп'ютерного моделювання процесів передавання даних через зашумлені канали зв'язку. Реалізацію моделювального середовища здійснено на підставі технології WPF (англ. *Windows Presentation Foundation*) та мови програмування C# для моделювання передачі даних із контрольованими параметрами завад.

Результати дослідження та їх обговорення / Research results and their discussion

Опис інтерфейсу розробленого програмного забезпечення. Для досягнення поставленої мети та виконання сформульованих завдань розроблено програмне забезпечення для моделювання процедури пересилання даних каналами зв'язку із завадами різними методами та аналізу їх властивостей. На рис. 1 зображено вигляд програми на початку її роботи. Програмне вікно містить чотири зони: одна для керування моделлю та три для відображення аналітичних даних моделі.

Рис. 1. Вигляд програмного забезпечення для моделювання процедури пересилання даних каналами зв'язку із завадами та аналізу властивостей методів кодування даних / View of the software for simulating the transmission of data through a communication channel with interference and analyzing the properties of data coding methods for such transmission

Проаналізуємо детальніше першу зону головного вікна. Зверху розташовано поля для введення числових даних:

- кількість спроб мутацій – визначає скільки разів канал зв'язку буде робити спробу застосувати завади і мутувати, тобто змінити, повідомлення.
- шанс мутації – число від 0 до 1, яке позначає ймовірність виникнення мутації, тобто зміни, в повідомленні під час його пересилання каналом зв'язку (значення 1 відповідає 100 % ймовірності).
- кількість пакетів даних (звичайних, кодів Ріда-Соломона, ІКВ-кодів) – визначає скільки пакетів даних буде надіслано кожним з методів. Корисне навантаження, яке передається кожним методом, є однакового розміру; відрізняється тільки надлишковістю і, як наслідок, остаточний розмір повідомлення, передаване каналом зв'язку.

Нижче числових полів розташований ряд опцій:

- надсилати дані (звичайні, коди Ріда-Соломона, ІКВ-кодів) – визначає, чи в процесі моделювання будуть надсилатись коди певного виду. Ця група прапорців також вмикає відповідні числові поля.
- використовувати 3-вимірні ІКВ-коди, які дають змогу закодовувати дані у вигляді тривимірних ІКВ-кодів, а не двовимірних.

Рис. 2. Інформаційне вікно / Information window

Далі є дві кнопки:

- почати – розпочинає моделювання, використовуючи згадані вище параметри. Під час моделювання збираються аналітичні дані, які відображаються в інших трьох зонах головного вікна програми. Про них нижче.
- інформація – відкриває інформаційне вікно з додатковою інформацією про параметри моделі (рис. 2).

Одразу ж після ряду кнопок розташований динамічний текст, який відображає процедуру моделювання у відсотках (під час її виконання). Натиснувши кнопку "Почати", розпочинається моделювання, під час якого динамічно оновлюються аналітичні дані та прогрес. Для кожного обраного методу кодування запускається

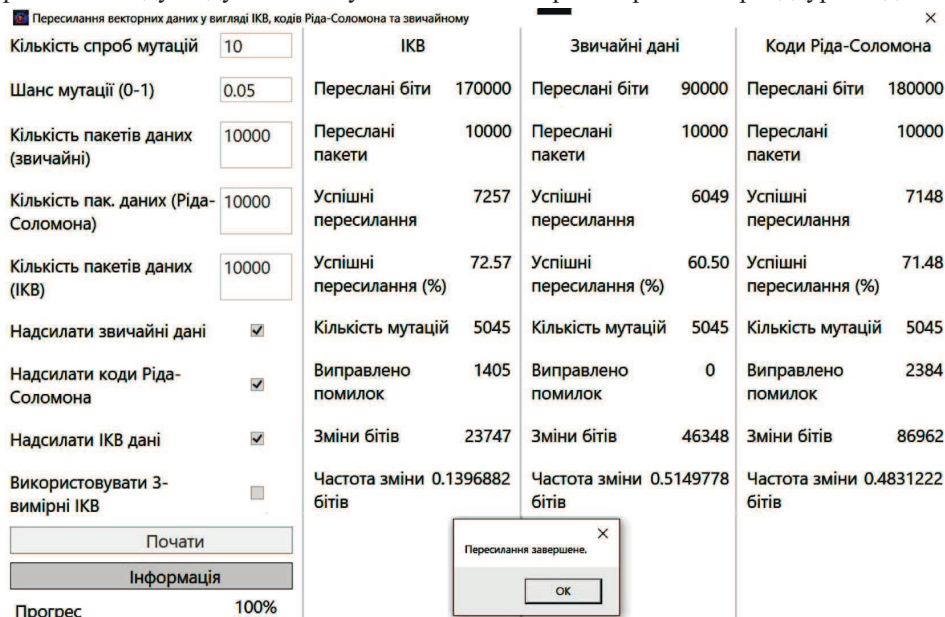


Рис. 3. Результат моделювання та повідомлення про це / Results of the simulation and notification

Тепер детальніше про три аналітичні зони. Очевидно, що кожна з цих зон пов'язана з певним методом кодування даних: коди МКВ, коди Ріда-Соломона, звичайні дані. Кожна зона має вісім динамічних текстових полів, серед яких:

- переслані біти – загальна кількість бітів, які надіслано за весь час процедури моделювання з використанням окремого методу кодування;
- переслані пакети – загальна кількість пакетів даних (які є корисним навантаженням), які було надіслано впродовж процедури моделювання;
- успішні пересилання – кількість пакетів, які було успішно переслано через канал зв'язку, враховуючи також й ті, що зазнали мутації, але були в подальшому виправлені (у випадку кодів МКВ та кодів Ріда-Соломона);
- успішні пересилання (%) – частка успішних пересилань у відношенні до загальної кількості пересланих пакетів;
- кількість мутацій – загальна кількість ситуацій, коли у каналі зв'язку відбулась мутація повідомлення через змодельовані завади;
- виправлено помилок – кількість пакетів, у яких вдалось виправити усі помилки;
- зміни бітів – загальна кількість ситуацій, коли в каналі зв'язку під час передачі даних змінювались біти на протилежне значення (з нуля на одиницю та навпаки) впродовж процедури моделювання;
- частота зміни бітів – відношення кількості ситуацій, коли в каналі зв'язку змінилось значення сигналу на протилежне, до загальної кількості пересланих бітів у цьому каналі (число від 0 до 1, де 0 означає жодних змін, тобто надсилаються тільки нулі або одиниці, а 1 означає, що кожен пересланий біт каналом не дорівнює своєму попереднику, тобто значення сигналу змінюється максимально часто).

Програмне забезпечення є простим у роботі, зручним у використанні, а його інтерфейс є лаконічним й інтуїтивним для користувача. З точки зору продуктивності, програмний застосунок не має жодних візуальних зависань, швидко відгукується на дії користувача, а

власний процес моделювання в окремому потоці, і, залежно від кількості обраних методів, одночасно може працювати до трьох обчислювальних потоків і один основний потік для відображення користувацького інтерфейсу та взаємодії з ним. Наприкінці програма повідомляє про завершення процедури моделювання (рис. 3).

результати формуються дуже швидко: для зображених на рис. 3 даних (10 тис. пакетів для усіх трьох видів кодування) програмний застосунок дав результати вже за 3 с після початку процедури моделювання.

Архітектура програмного застосунку. Проаналізуємо діаграму UML-класів для розробленого програмного застосунку (рис. 4) для кращого розуміння, як він влаштований всередині. Бачимо, що застосунок поділено на три модулі:

- View – відповідає за відображення та взаємодію з користувацьким інтерфейсом.
- Test – модуль з Unit-тестами підмодулів програмного застосунку та найчастіших сценаріїв його використання. Призначений для того, щоб бути певними, що всі підмодулі працюють справно.
- Core – головний модуль, що містить основну логіку програмного застосунку.

Модуль Core (укр. ядро) становить найбільшу цікавість для аналізу. Зі схеми видно, що під час розроблення програмного забезпечення дотримувались одних з найважливіших принципів проектування систем – SOLID (об'єктно-орієнтоване програмування). У своїй книзі [9] відомий та авторитетний автор Роберт С. Мартін (дядечко Боб) наголошує на важливості дотримання принципів SOLID як основних рекомендацій для написання підтримуваного, масштабованого та гнучкого об'єктно-орієнтованого коду. Ці принципи допомагають розробникам створювати зрозумілий, тестований і розширюваний код, що відповідає практикам гнучкої реалізації програмного проекту. Усі спроектовані класи відповідають принципам SOLID [9]:

- Single Responsibility Principle (SRP) – класи виконують тільки одну конкретну задачу, тому мають тільки одну причину для змін;
- Open/Closed Principle (OCP) – код відкритий для розширення, але закритий для модифікації, так уникаємо небажаних змін;

- Liskov Substitution Principle (LSP) – об'єкти підкласів коректно замінюють об'єкти базового класу без зміни очікуваної поведінки;
- Interface Segregation Principle (ISP) – не використовуємо великих інтерфейсів;
- Dependency Inversion Principle (DIP) – вищі рівні системи не залежать від нижчих рівнів, а обидва залежать від встановлених абстракцій.

Це чудово продемонстровано на схемі тим, що не знаходиться в основному модулі Core жодного конкретного класу, який залежав би він інших конкретних класів. Всі вони залежать від абстракцій, а ці абстракції також залежать від інших абстракцій.

На схемі також можна побачити, що система використовує концепцію ІEncoder-ів, тобто кодувальників, та ІDecoder-ів, тобто декодувальників (розкодувальників). Перші відповідають за те, щоб перетворити пакети даних на їх закодовані певним методом версії. Ці дані зберігаються у повідомленнях ІMessage. Другі ж відповідають за те, щоб розкодувати отриману інформацію і отримати вхідні дані (у процесі цього, можуть бути виправлені помилки). Також бачимо абстрактні сутності ІMessageSource та ІMessageDestination, які поєднані між собою за допомогою ІPipe у канал зв'язку, який піддається завадам, що регулюються сутністю ІMutationLogic.

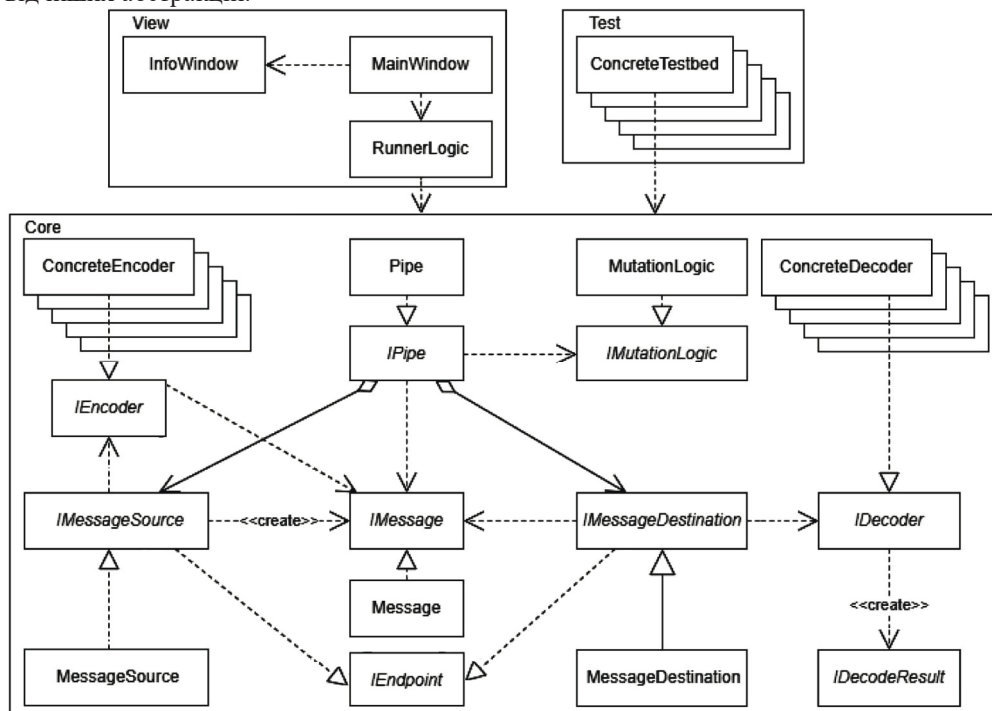


Рис. 4. Діаграма UML-класів для загальної архітектури програмного застосунку / UML class diagram for the overall software application

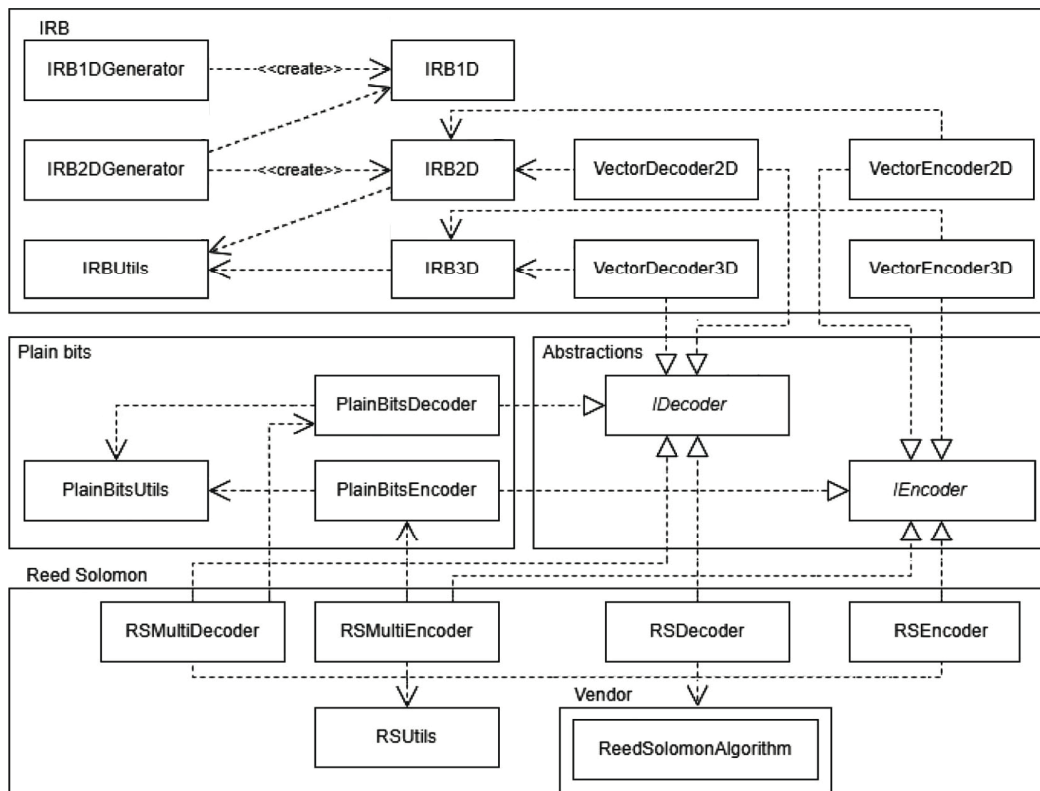


Рис. 5. Діаграма UML-класів для сегменту архітектури, пов'язаного з кодуванням та розкодуванням / UML class diagram for the architecture segment related to encoding and decoding

В абстрактних сутностей є конкретні реалізації. А в кодувальників і декодувальників цих реалізацій є багато. Зв'язки цих сутностей детальніше можна побачити в наступній діаграмі (рис. 5). Тут бачимо розділення на підмодулі. Virізняється серед них Abstractions. Цей підмодуль є вищого рівня. Інші три підмодулі: IRB (відповідає за коди МІКВ), Plain bits (відповідає за звичайні дані) та Reed Solomon (відповідає за коди Ріда-Соломона) – є конкретними реалізаціями, є нижчого рівня, і тому мають направлені стрілки до цього підмодуля абстракцій. Недопустимо мати хоча б одну стрілку з компонента вищого рівня до компонента нижчого рівня. У такому випадку порушувались би принципи проектування чистої архітектури, які описував той ж дядечко Боб [8]. Бачимо, що в кожному компоненті є реалізація абстрактних IDecoder та IEncoder. У компоненті IRB також є допоміжні класи для роботи з кодами МІКВ. У компоненті Reed Solomon є дві пари реалізацій (де) кодувальників. Одна пара відповідає за роботу з одинарним пакетом даних, а інша (RSMultiDecoder та RSMultiEncoder) – для роботи з кількома пакетами водночас. Також у цьому компоненті є використання коду сторонніх авторів з команди ZXing (вони відмінно реалізували кодування та декодування кодів Ріда-Соломона).

Для подальших пояснень важливо вказати, що для найпростішого тесту використовуємо пересилання даних, які кодують максимум 273 унікальні значення. 273 – це S для кодів МІКВ з $n = 17$ та $R = 1$. А також 273 унікальні значення можна закодувати за допомогою звичайної двійкової системи числення 9-ма бітами.

Хочемо детальніше пояснити процедуру кодування декількох пакетів водночас. Справа в тому, що алгоритм для кодування Ріда-Соломона не може працювати з малими розмірами повідомлень (наприклад з тими ж 9-ма бітами). Тому, кодуючи тільки один пакет даних, доводилось би додавати багато зайвих бітів. Тому було прийнято рішення використовувати повідомлення розміром 9 байтів, 4 з яких є надлишковими для можливості виправлення 2 помилок, і 5 байтів корисного навантаження. При цьому, щоб використати це доступне навантаження максимально ефективно, в одному повідомленні пересилається 4 пакети по 9 біт кожен. Для виправлення проблеми появи таких помилок використано вже згаданий мажоритарний принцип. І наостанок, зазначимо, що в модулі тестів було розроблено 13 Unit-тестів для підтримки коректної роботи програми.

Результати проведеного експерименту. Було реалізовано можливість пересилати дані, використовуючи двовимірні та тривимірні МІКВ-коди. Обидва види кодів мають однаковий час на кодування даних і їх декодування (табл. 1). Оброблення даних із звичайною двійковою системою числення очікувано займає найменше часу. Використання кодів Ріда-Соломона є на 221 % швидшим, ніж за допомогою МІКВ-кодів. Тривалість зростає з лінійною залежністю від кількості пакетів даних.

Табл. 1. Порівняння загальної тривалості пересилання 1 млн пакетів даних (з кодуванням, мутацією та декодуванням) / Comparison of the total time for sending a message (including encoding, mutation, and decoding)

	Запуск 1, мс	Запуск 2, мс	Запуск 3, мс	Середнє, мс
2-вимірні ІКВ-коди	8732	8766	8868	8789
3-вимірні ІКВ-коди	8502	8803	8819	8708
Звичайні дані	897	873	905	892
Коди Ріда-Соломона	2828	2654	2703	2728

Що ж до надлишковості, то для кодів МІКВ є дещо кращим методом порівняно з кодами Ріда-Соломона (табл. 2): 47 проти 50 %. Збільшення кількості помилок, що можуть виправити коди Ріда-Соломона, тримаючи довжину повідомлення якомога меншою в допустимих межах, не призводить до зміни надлишковості, але очевидно збільшує шанси на успішні пересилання (проте шанси успішних пересилань зменшуються). Але йдеться про величезні повідомлення, тож обмежимося тільки наближеними до найменш можливого значення, щоб порівняння з кодів МІКВ (17 бітів) було більш справедливим. Коди МІКВ різної вимірності не порівнюємо, бо для них результати ідентичні.

Табл. 2. Порівняння надлишковості методів кодування / Comparison of redundancy of coding methods

	Пересла- но, біт	Біт / пак- ет	Надлишко- вість		Успішні пе- ресилан- ня, %
			біт	%	
ІКВ-коди	170000	17	8	47	73
Звичайні дані	90000	9	0	0	60
Коди Р.-С. (2, 9)	180000	18	9	50	74
Коди Р.-С. (2, 16)	128000	13	4	31	70
Коди Р.-С. (2, 24)	113088	11	2	18	69
Коди Р.-С. (3, 13)	173368	17	8	47	90
Коди Р.-С. (3, 24)	120000	12	3	25	88
Коди Р.-С. (4, 17)	170000	17	8	47	97

Примітка: пересилається 10 тис. пакетів даних. Для кодів Ріда-Соломона в дужках вказано кількість помилок, які можна виправити, а також загальна кількість байтів повідомлення (з корисним і надлишковим навантаженням).

Далі розглянемо залежність частки успішних пересилань пакетів даних залежно від того, наскільки сильні завади (рис. 7). З графіка бачимо, що звичайне кодування двійковою системою числення поступається кодам МІКВ. Також можна помітити, що ці два види кодування втрачають ефективність експоненційно. Коди ж Ріда-Соломона мають графік схожий на половинку нормального розподілу. До певного моменту коди Ріда-Соломона зберігають ефективність завдяки повному розкриттю своїх властивостей виправлення помилок. Чітко це помітно завдяки графіку на рис. 8 – на ньому яскраво виражений екстремум в районі ймовірності мутації 0.04. Розташування точки перелому залежить від кількості надлишкових бітів для виправлення помилки.

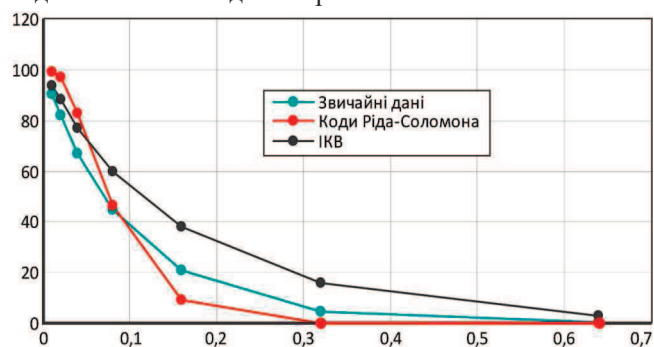


Рис. 6. Порівняння частки успішних пересилань (вертикальна вісь) 10 тис. пакетів для різних методів залежно від шансу мутації (горизонтальна вісь) за кількості спроб – 10 / Comparison of the percentage of successful transfers (vertical axis) of 10000 packets for different methods depending on the chance of mutation (horizontal axis) with the number of attempts – 10

На графіку (рис. 8) порівняння кількості виправлених помилок має схожість до певної сигмоїдної залежності, коли ефективність спершу зростає, а потім падає

через певні фактори (в даному випадку – через досягнення певного порогу, після якого ефективність кодів з такою надлишковістю активно починає зменшуватись).

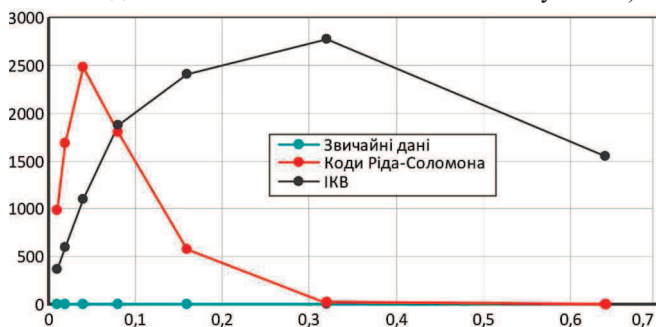


Рис. 7. Порівняння кількості виправлених помилок (вертикальна вісь) за пересилання 10 тис. пакетів для різних методів залежно від шансу мутації (горизонтальна вісь) за кількості спроб – 10 / Comparison of the number of corrected errors (vertical axis) when transferring 10,000 packets for different methods depending on the mutation chance (horizontal axis) with the number of attempts – 10

Також розглянемо залежності від зміни розрядності (n) кодів МКВ. З рис. 8 бачимо, що зі збільшенням розрядності кодів МКВ, час на оброблення зростає експоненційно. На рис. 9 помітно, що зі збільшенням розрядності, успішність пересилання даних зростає логарифмічно. Кількість виправлених помилок теж зростає зі збільшенням розрядності кодів МКВ (рис. 10). І очевидно, що від збільшення розрядності (n) зменшується також частота зміни бітів (рис. 11).

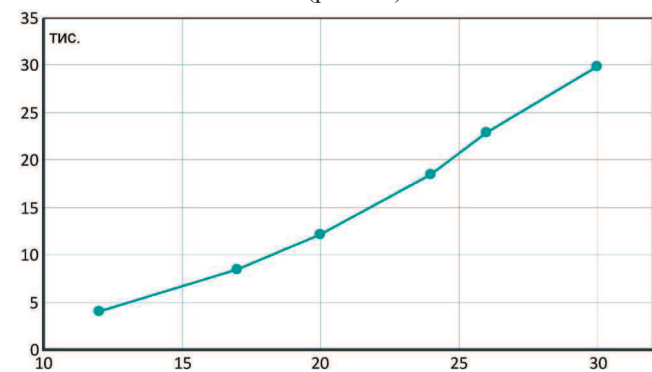


Рис. 8. Залежність тривалості оброблення даних (в мс, 1 млн пакетів) за допомогою МКВ-кодів від її розрядності (горизонтальна вісь) за 10 спроб мутації з шансом 0,05 / Dependence of the data processing time (in milliseconds, 1 mln packets) using MIRB codes on its bit count (horizontal axis) with 10 mutation attempts with a chance of 0.05

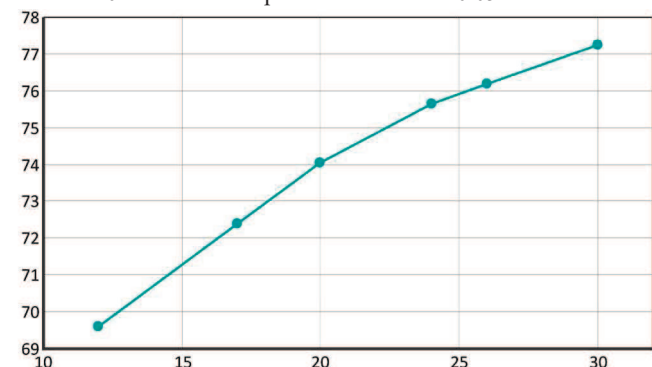


Рис. 9. Залежність успішності пересилання (у %, 1 млн пакетів) за допомогою МКВ-кодів від її розрядності (горизонтальна вісь) за 10 спроб мутації з шансом 0,05 / Dependence of transferring success (in percentage, 1 mln packets) using MIRB codes on its bit count (horizontal axis) with 10 mutation attempts with a chance of 0.05

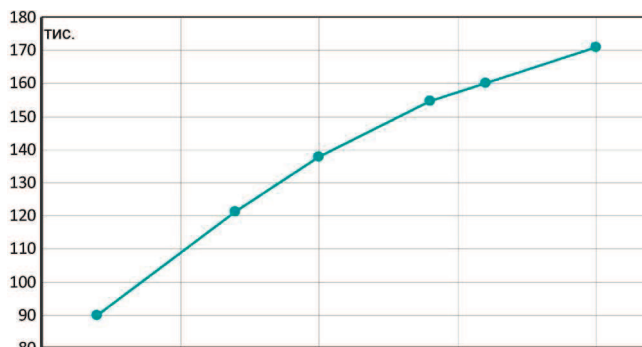


Рис. 10. Залежність кількості виправлених помилок при пересиланні 1 млн пакетів МКВ-кодами від її розрядності (горизонтальна вісь) за 10 спроб мутації з шансом 0,05 / Dependence of the number of corrected errors when transferring 1 mln packets using MIRB codes on its bit count (horizontal axis) with 10 mutation attempts with a chance of 0.05

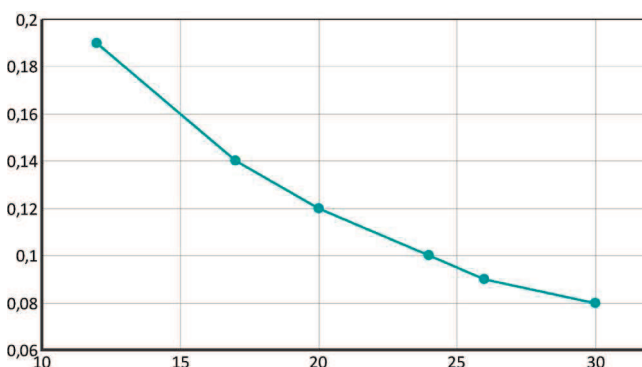


Рис. 11. Залежність частоти зміни бітів при пересиланні 1 млн пакетів МКВ-кодами від її розрядності (горизонтальна вісь) за 10 спроб мутації з шансом 0,05 / Dependence of the bit flipping rate when transferring 1 mln packets using MIRB codes on its bit count (horizontal axis) with 10 mutation attempts with a chance of 0.05

До вашої уваги також графік на рис. 12, який відображає залежність частоти виправлення помилок до розмірності МКВ-кодів за стабільної кількості завад у каналі зв'язку.

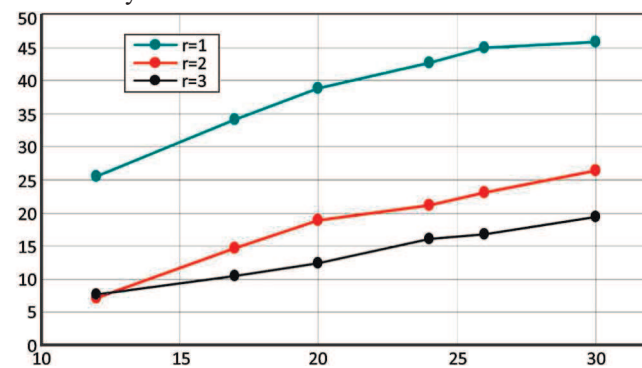


Рис. 12. Залежність частки виправлених помилок до розмірності МКВ-кодів при пересиланні 10 тис. пакетів за чіткої кількості мутацій r / Dependence of the percentage of corrected errors on the MIRB codes bit count when transferring 10,000 packets with a fixed number of mutations r

З результатів, отриманих у межах спостережень проведеного експерименту, робимо такі висновки. Процес оброблення кодів МКВ займає втричі більше процесорного (табл. 1), що є їхнім недоліком.

Коди МКВ є менш надлишковими, ніж коди Ріда-Соломона (табл. 2). Це не є великою перевагою. Проте коди МКВ, на відміну від кодів Ріда-Соломона, можуть кодувати невеликі обсяги даних (наприклад 9 бі-

тів, коли $n = 17$). Коди Ріда-Соломона ж вимагають більших розмірів даних, до того ж розміри мають бути виражені цілою кількістю байт (мінімум 5 байт, тобто 40 біт). Тому доводилось пакувати декілька пакетів в одне повідомлення (для формування одного повідомлення з чотирьох пакетів з $n = 17$ потрібно 72 біти), щоб уникнути величезної надмірності в іншому випадку. Це є перевагою кодів МІКВ над кодами Ріда-Соломона.

Зі збільшенням частоти появи помилок, коди Ріда-Соломона швидко досягають точки, коли вони не справляються з виявленням помилок. Для кодів МІКВ ця точка настає набагато пізніше (рис. 6 і 7). У більш зашумлених каналах зв'язку МІКВ-коди справляються набагато краще. Тривалість оброблення даних з використанням кодів МІКВ зростає експоненційно (рис. 8), тож МІКВ-коди малої розмірності мають велику перевагу над МІКВ-кодами великої розмірності. Проте зі збільшенням розмірності зростає частота успішних пересилань повідомлень (рис. 9) завдяки збільшенню можливостей виправлення помилок (рис. 10). Тому, зважаючи на ці два фактори, потрібно знайти баланс між ними і, залежно від вимог до проєктованого програмного застосунку, визначити потрібну їхню розмірність. Окрім цього, зі збільшенням розрядності, зменшується й частота зміни бітів (рис. 11).

За будь-якої розглянутої розмірності МІКВ-кодів, вони мають меншу частоту зміни бітів (0,14) порівняно з кодами Ріда-Соломона. Для яких цей показник становить 0,48-0,49 залежно від параметрів. Тож МІКВ-коди є кращими приблизно у 3,5 рази. Для кодування звичайною двійковою системою числення цей показник становить 0,51. Зважаючи на згадані дослідження інших авторів [3, 6, 10], робимо висновок, що МІКВ-коди, завдяки своїй властивості зниження частоти зміни бітів, здатні зменшувати енергоспоживання програмно-апаратних систем.

Графік з рис. 12 дуже схожий до того, який отримала раніше згадана група науковців [16]. З різницею в тому, що в них частка виправлення помилок зростає приблизно вдвічі. Ця різниця частково пов'язана з тим, що в нашому програмному забезпеченні враховуємо тільки виправлені помилки в тих пакетах даних, які змогли бути виправлені повністю. Також ця різниця може вказувати на те, що автори використовували R відмінне від 1, що водночас збільшує частоту виявлення та виправлення помилок.

Отримані результати дослідження демонструють високу ефективність застосування МІКВ-кодів у зашумлених каналах зв'язку, особливо в умовах значних завад, де вони перевершують коди Ріда-Соломона за часткою успішних пересилань та енергоефективністю. Завдяки меншій надлишковості та здатності виправляти велику кількість помилок, МІКВ-коди забезпечують надійність передавання даних навіть у складних умовах. Отже, результати дослідження підтверджують доцільність використання запропонованого підходу в сучасних комунікаційних системах, де важливими є завадостійкість і енергоефективність.

Обговорення результатів дослідження. У роботі [15] розглянуто можливість використання ІКВ-кодів для дослідження завадостійких циклічних кодів і поліпшення їхніх параметрів на підставі використання властивостей ІКВ-кодів. Автори розробили програму для побудови циклічних і монолітних кодів на підставі ІКВ-

кодів і дослідження їхньої коректувальної здатності. Також автори запропонували спосіб збільшення потужності ІКВ-кодів.

У роботі [17] досліджено циклічні коди, побудовані на підставі ІКВ-кодів і розглянуто методи оптимального синтезу за допомогою ІКВ-кодів циклічних кодів з поліпшеними можливостями щодо виправлення багаторазових помилок. Також у цій роботі проаналізовано структурні відмінності між ІКВ-кодом та БЧХ-кодом. Окрім цього, встановлено, що теоретичні можливості ІКВ-кодів з виправлення помилок є набагато більшими, ніж в БЧХ-кодів, проте це досягається за рахунок ускладнення приймально-передавальної апаратури та уповільнення передавання.

У роботі [18] досліджено різновиди завадостійких циклічних кодів, побудованих за допомогою незвідних поліномів і нестандартних математичних конструкцій з унікальними комбінаційними властивостями – "ідеальних кільцевих в'язанок" (ІКВ-кодів); а також розглянуто методи оптимального синтезу за допомогою ІКВ-кодів циклічних кодів з поліпшеними можливостями щодо виправлення багаторазових помилок. Встановлено, що, на відміну від кодів БЧХ, для побудови ІКВ-кодів зі значним рівнем завадостійкості не потрібні складні обчислення, а допустимі довжини кодових комбінацій ІКВ-кодів дають змогу значно розширити спектр побудови кодів зі значним рівнем захищеності від завад.

У роботі [21] продемонстровано спосіб моделювання процедури передачі даних, закодованих за допомогою завадостійких кодів Ріда-Соломона на підставі об'єктно-орієнтованої технології. Встановлено, що алгебраїчні процедури кодування та декодування даних за допомогою цього коду є простими, ефективними та прозорими, а код характеризується, як потужний та зі значною коригувальною здатністю. У роботі автори використали ООП-підхід, який є провідним для розроблення ПЗ у сучасному світі.

У роботі [1] досліджено різницю в безпеці та енергоспоживанні кодів Ріда-Соломона та БЧХ-кодів. Коди БЧХ виявились більш енергоефективними, проте менш захищеними. Цікаво, що найбільший розрив в енергоефективності є в каналах зв'язку з низьким рівнем завад.

Обговорення результатів дослідження підтвердило доцільність проведення саме цього дослідження, оскільки в аналізованій літературі відсутні детальні емпіричні порівняння МІКВ-кодів з кодами Ріда-Соломона щодо їхньої енергоефективності та завадостійкості. Отримані результати заповнюють цю наукову прогалину, демонструючи переваги МІКВ-кодів у специфічних умовах зашумлених каналів. Це вказує на новизну проведеного дослідження та його значущість для подальшого розвитку методів завадостійкого кодування.

Отже, внаслідок виконаної роботи можна сформулювати такі наукову новизну та практичну значущість результатів дослідження.

Наукова новизна отриманих результатів дослідження – вперше проведено емпіричне порівняння ефективності МІКВ-кодів порівняно з кодами Ріда-Соломона в умовах зашумлених каналів зв'язку щодо їхньої енергоефективності, надлишковості та здатності виправляти помилки, де краща енергоефективність забезпечується завдяки меншій частоті зміни бітів і більшою часткою успішних пересилань у середовищах із високим рівнем завад.

Практична значущість результатів дослідження – результати дослідження можна використати для обґрунтування вибору підходу, який є кращим для певного завдання передавання закодованих даних каналом зв'язку із врахування можливих завад.

Висновки / Conclusions

Проаналізовано ефективність використання монолітних ІКВ-кодів для кодування даних порівняно з кодами Ріда-Соломона, що дало змогу визначити їхні переваги та недоліки під час передачі закодованих даних каналами зв'язку. За результатами проведеного дослідження можна зробити такі основні висновки.

1. Розроблено програмне забезпечення для моделювання процедури пересилання повідомлень каналами зв'язку із завадами, використовуючи МІКВ-коди та коди Ріда-Соломона, яке дало можливість провести емпіричне порівняння методів кодування, а також підтвердити їх ефективність у різних умовах зашумлених каналів.
2. Встановлено низку переваг і недоліків кодування даних монолітними ідеальними кільцевими в'язанками (МІКВ) порівняно з іншими видами кодування, як, наприклад, кодами Ріда-Соломона. Серед переваг присутня дещо менша надлишковість, можливість працювати з даними малих розмірностей, більша енергоефективність за рахунок меншої частоти перемикачів бітів, здатність виправляти дуже велику кількість помилок, особливо після збільшення розмірності. Серед недоліків є більший час на оброблення даних і менша (ніж у кодів Ріда-Соломона) здатність виправляти помилки в каналах з малою кількістю завад. Ці висновки отримано на підставі даних, отриманих внаслідок проведеного експерименту.
3. З'ясовано, що наявні властивості монолітних ІКВ-кодів роблять їх особливо привабливими для використання у зашумлених каналах зв'язку, космічних технологіях, а також в IoT-системах, коли маємо справу з дуже короткими повідомленнями, значними вимогами до енергоефективності, а також з великою кількістю завад.

У майбутньому плануємо також порівняти популярні коди Ріда-Соломона з монолітними ІКВ-кодами з R, відмінним від 1; дослідити переваги пересилання даних ще більших розмірностей; знайти методи пришвидшення оброблення монолітних ІКВ-кодів.

References

1. Ahmed, M., Albashier, M., Aziz, A. A., Ghani, H. A., & Samingan, A. K. (2018). Performance comparison of energy efficiency and physical layer security for Reed-Solomon and Bose-Chaudhuri-Hocquenghem codes in wireless sensor networks. *2018 7th International Conference on Computer and Communication Engineering (ICCCCE)*, pp. 157–160. <https://doi.org/10.1109/icccce.2018.8539284>
2. Berrou, C., Glavieux, A., & Thitimajshima, P. (1993). Near Shannon limit error-correcting coding and decoding: Turbo-codes. *Proceedings of ICC 93 – IEEE International Conference on Communications*, 1064–1070. <https://doi.org/10.1109/icc.1993.397441>
3. Bittman, D., Gray, M., Raizes, J., Mukhopadhyay, S., Bryson, M., Alvaro, P., Long, D. D., & Miller, E. L. (2018). Designing Data Structures to Minimize Bit Flips on NVM. *2018 IEEE 7th Non-Volatile Memory Systems and Applications Symposium (NVMsA)*, 85–90. <https://doi.org/10.1109/nvmsa.2018.00022>
4. Boiko, J., & Pyatin, I. (2023). Features of code redundancy formation in information transmission channels. *Infocommunication and Computer Technologies*, 2(04), 12–25. <https://doi.org/10.36994/2788-5518-2022-02-04-01>
5. Bose, R. C., & Ray-Chaudhuri, D. K. (1960). On a class of error correcting binary group codes. *Information and Control*, 3(1), 68–79. [https://doi.org/10.1016/s0019-9958\(60\)90287-4](https://doi.org/10.1016/s0019-9958(60)90287-4)
6. Fallin, A., & Burtscher, M. (2023). Reducing Memory-Bus Energy Consumption of GPUs via Software-Based Bit-Flip Minimization. *2022 IEEE/ACM Workshop on Memory Centric High Performance Computing (MCHPC)*, 21–29. <https://doi.org/10.1109/mchpc56545.2022.00008>
7. Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2), 147–160. <https://doi.org/10.1002/j.1538-7305.1950.tb00463.x>
8. Martin, R. C. (2018). Clean architecture: A craftsman's guide to software structure and Design. *Prentice Hall*. URL: https://books.google.com.ua/books/about/Clean_Architecture.html?id=8ngAkAEACAAJ&redir_esc=y
9. Martin, R. C., & Martin, M. (2006). Agile principles, patterns, and practices in C#. *Pearson Education*. URL: https://www.google.com.ua/books/edition/Agile_Principles_Patterns_and_Practices/hck7v6g09oC?gbpv=0
10. Pasko, I., & Shchenyakin, O. (2014). Study of energy efficiency of linear block codes with improved properties. *Information processing systems*, 2, 134–136. [In Ukrainian]. URL: http://nbuv.gov.ua/UJRN/soi_2014_2_29
11. Riznyk, V. (1989). Synthesis of optimal combinatorial systems. *Vyshcha shkola*. [In Ukrainian]. URL: https://scholar.google.com.ua/citations?view_op=view_citation&hl=uk&user=Lhjxr1MAAAAJ&citation_for_view=Lhjxr1MAAAAJ:YAc2jw7qUC
12. Riznyk, V. (2015). Optimal codes on vector combinatorial configurations. *Journal of Lviv Polytechnic National University. Information systems and networks*, 814(1), 130–138. [In Ukrainian]. URL: http://nbuv.gov.ua/UJRN/VNULPICM_2015_814_15
13. Riznyk, V. (2022). Combinatorial optimization of systems of neural network cryptographic data protection. *Ukrainian Journal of Information Technology*, 4(2), 56–60. <https://doi.org/10.23939/ujit2022.02.056>
14. Riznyk, V., & Skrybaylo-Leskiv, D. (2011). Synthesis of interference-resistant codes based on multi-position ideal ring bundles. *Computer sciences and information technologies*, 694, 199–205. [In Ukrainian]. URL: <https://science.lpnu.ua/scsit/all-volumes-and-issues/volume-694-2011/sintez-zavadostykykh-kodiv-na-osnovi-bagatopozyciynih>
15. Riznyk, V., Pylyp, V., & Tymchenko, T. (2012). Investigation of noise-resistant cyclic codes on combinatorial configurations. *Computer sciences and information technologies*, 732, 339–342. [In Ukrainian]. URL: <https://science.lpnu.ua/scsit/all-volumes-and-issues/volume-732-2012/doslidzhennya-zavadostykykh-cyklichnykh-kodiv-na>
16. Riznyk, V., Skrybaylo-Leskiv, D., Badz, B., Hlod, S., Liakh, V., Kulyk, Y., Romanjuk, N., Tkachuk, K., & Ukrainets, V. (2021). Comparative analysis of monolithic and cyclic noise-protective codes effectiveness. *Ukrainian Journal of Information Technology*, 3(1), 99–105. <https://doi.org/10.23939/ujit2021.03.099>
17. Riznyk, V., Skrybaylo-Leskiv, D., & Lyakhovych, O. (2009). Structural analysis of methods for optimal construction of cyclic codes with improved characteristics. *Computer sciences and information technologies*, 650, 190–194. [In Ukrainian]. URL: <https://science.lpnu.ua/uk/scsit/vsi-vypusky/vypusk-650-2009/strukturyy-analiz-metodiv-optymalnoyi-pobudovy-cyklichnykh-kodiv-z>
18. Riznyk, V., Skrybaylo-Leskiv, D., Lyakhovych, O., & Yurchak, I. (2010). Comparative analysis of noise immunity of multi-position cyclic codes. *Computer-aided design systems. Theory and practice*, 685, 147–151. [In Ukrainian]. URL: <https://science.lpnu.ua/cds-archive/all-volumes-and-issues/number-685-2010/porivnyalnyi-analiz-metodiv-optimalnoyi-pobudovy-cyklichnykh-kodiv-z>
19. Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3), 379–423. <https://doi.org/10.1002/j.1538-7305.1948.tb01338.x>
20. Tucker, R. S. (2012). Energy consumption in telecommunications. *2012 Optical Interconnects Conference*, 1–2. <https://doi.org/10.1109/oic.2012.6224478>
21. Vavruk, E., Popovych, B., & Popovych, R. (2021). Program model of Reed-Solomon Codes. *Computer Systems and Networks*, 3(1), 1–6. <https://doi.org/10.23939/csn2021.01.001>

22. Velyka, O. (2008). Synthesis of ring monolithic codes based on the method of graph transformations of cyclic groups of Galois fields. *Computer sciences and information technologies*, 616,

190–193. [In Ukrainian]. URL: <https://science.lpnu.ua/scsit/all-volumes-and-issues/volume-616-2008/sintez-kilcevih-monolitnih-kodiv-na-osnovi-metodu>

Yu. Yu. Bilas, V. V. Riznyk

Lviv Polytechnic National University, Lviv, Ukraine

ANALYSIS OF THE EFFICIENCY OF USING MONOLITHIC IRB CODES IN COMPARISON WITH REED-SOLOMON CODES

Ensuring error-resistant and efficient encoding is crucial in modern communication and data processing systems, particularly in high-noise environments with limited resources. Data transmission reliability directly affects network performance and energy consumption, making encoding techniques a key factor in system optimization. Therefore, in the course of our research, comparison of the performance of monolithic ideal ring bundles (IRB) and Reed-Solomon codes for data transmission over noisy communication channels, focusing on error correction capabilities, redundancy, energy efficiency, and computational complexity, has been done. A software-based simulation system has been developed in C# and WPF, enabling controlled testing of different noise levels and interference patterns while evaluating processing costs, error correction efficiency, redundancy, and bit transition frequency. It has been demonstrated that IRB-based codes offer superior error correction under severe interference while reducing bit transitions by approximately 3.5 times compared to Reed-Solomon codes, resulting in significant energy savings. This property is especially beneficial for applications where minimizing energy consumption is critical, such as battery-powered IoT devices and satellite communication systems. However, the major drawback of IRB codes is found to be their significantly higher computational cost, which may limit their use in high-speed networks with strict latency requirements. The increased processing complexity poses challenges for real-time or low-latency applications, necessitating further optimization of encoding and decoding algorithms. Moreover, IRB codes are demonstrated to have a unique ability to correct a very high number of errors in a non-deterministic manner, making them particularly useful in large-scale data transmission over highly noisy channels. As a consequence, this property allows IRB-based codes to outperform traditional approaches in environments with dynamic noise levels. These advantages make IRB codes a promising choice for mobile and satellite communication systems, as well as IoT devices, where energy efficiency and adaptability to varying channel conditions are essential. Future research should focus on optimizing IRB encoding algorithms to reduce computational complexity, enhance processing speed, and explore their potential integration into next-generation wireless and space communication technologies.

Keywords: ideal ring bundles; encoding; noise resistance; energy efficiency.